

José Milton de Araújo

**PROGRAMAS PARA
DIMENSIONAMENTO
E VERIFICAÇÃO DE
CONCRETO ARMADO**

Editora Dunas

1ª edição: 2018

Fortran 90

Visual Basic 2017

Visual C# 2017

**PROGRAMAS PARA DIMENSIONAMENTO
E VERIFICAÇÃO DE CONCRETO ARMADO**

JOSÉ MILTON DE ARAÚJO

**PROGRAMAS PARA DIMENSIONAMENTO
E VERIFICAÇÃO DE CONCRETO ARMADO**

Editora DUNAS

PROGRAMAS PARA DIMENSIONAMENTO E VERIFICAÇÃO DE CONCRETO ARMADO

© Copyright Editora DUNAS

Araújo, José Milton de
Programas para dimensionamento e verificação de concreto armado / José Milton de Araújo. – 1. ed. –
Rio Grande, RS : Editora Dunas, 2018.

Bibliografia
ISBN 978-85-86717-19-2

1. Concreto armado. I. Título

CDU 624.012.45
CDD 624.1834

Editora DUNAS
Rua Tiradentes, 105 - Cidade Nova
96211-080 RIO GRANDE - RS - Brasil

www.editoradunas.com.br
e-mail: contato@editoradunas.com.br

1ª edição, Setembro/2018
Reimpressão: Agosto/2021

APRESENTAÇÃO

Este livro é um complemento da obra Curso de Concreto Armado. Nessa coleção foram apresentados, dentre tantos assuntos, os procedimentos de dimensionamento e verificação de seções de concreto armado. Geralmente, ao final de cada capítulo de dimensionamento, foi apresentado um roteiro para organizar o trabalho de cálculo realizado manualmente. Esses roteiros já são, praticamente, roteiros de programação e têm sido utilizados por muitos leitores para a elaboração dos seus próprios programas.

Neste livro, são apresentados alguns programas para dimensionamento e verificação de seções de concreto armado. Consideram-se os casos de dimensionamento e verificação de seções retangulares e seções T sob flexão normal simples, dimensionamento ao esforço cortante, dimensionamento e verificação de seções retangulares sob flexo-compressão normal, dimensionamento à torção e à flexo-tração.

Para cada um desses temas, são fornecidas as listagens dos códigos-fonte nas linguagens Fortran 90, Visual Basic 2017 e Visual C# 2017. O leitor que não utiliza nenhuma dessas linguagens de programação não encontrará dificuldades em adaptar os programas para a linguagem de sua preferência. Para isto, basta seguir o roteiro de programação que é apresentado no início de cada capítulo.

No momento do lançamento dessa reimpressão, a Microsoft disponibiliza o Visual Studio 2019. As instruções para download apresentadas no capítulo 1 permanecem inalteradas. Assim, basta o leitor fazer o download gratuito do Microsoft Visual Studio Community 2019 e utilizar os programas aqui apresentados sem nenhuma modificação.

O objetivo deste livro é apresentar uma sequência de programação simples, que permite resolver os problemas mencionados, porém, sem nenhuma preocupação com os preciosismos técnicos das linguagens de programação. O mais importante é que o leitor, previamente encantado com o concreto armado, também se interesse pelo desenvolvimento de ferramentas numéricas de cálculo. Espero que este livro consiga trazer esse estímulo aos seus leitores.

José Milton

SUMÁRIO

1. INTRODUÇÃO	1
1.1 - Organização do livro	1
1.2 - As linguagens de programação utilizadas	2
1.3 - Entrada de dados e separador decimal	2
1.4 - Preparando o ambiente de trabalho	3
1.5 - Melhoramentos dos programas	6
1.6 - O software PACON	6
2. FLEXÃO NORMAL SIMPLES	
Dimensionamento de Seções Retangulares	9
2.1 - Apresentação do problema	9
2.2 - Roteiro para programação	9
2.3 - Programa em Fortran 90	13
2.4 - Programa em Visual Basic 2017	16
2.5 - Programa em Visual C# 2017	21
2.6 - Exemplos para testar os programas	26
3. FLEXÃO NORMAL SIMPLES	
Dimensionamento de Seções T	29
3.1 - Apresentação do problema	29
3.2 - Roteiro para programação	29
3.3 - Programa em Fortran 90	33
3.4 - Programa em Visual Basic 2017	37
3.5 - Programa em Visual C# 2017	43
3.6 - Exemplos para testar os programas	49
4. FLEXÃO NORMAL SIMPLES	
Verificação da Capacidade Resistente	51
4.1 - Apresentação do problema	51
4.2 - Roteiro para programação	51
4.3 - Programa em Fortran 90	55
4.4 - Programa em Visual Basic 2017	58
4.5 - Programa em Visual C# 2017	63
4.6 - Exemplos para testar os programas	69
5. ESFORÇO CORTANTE	71
5.1 - Apresentação do problema	71
5.2 - Roteiro para programação	71
5.3 - Programa em Fortran 90	73
5.4 - Programa em Visual Basic 2017	76
5.5 - Programa em Visual C# 2017	80
5.6 - Exemplos para testar os programas	83
6. FLEXO-COMPRESSÃO NORMAL	
Dimensionamento de Seções Retangulares	85
6.1 - Apresentação do problema	85

6.2 - Roteiro para programação	85
6.3 - Programa em Fortran 90	91
6.4 - Programa em Visual Basic 2017	97
6.5 - Programa em Visual C# 2017	104
6.6 - Exemplos para testar os programas	113
 7. FLEXO-COMPRESSÃO NORMAL	
Verificação de Seções Retangulares	115
 7.1 - Apresentação do problema	115
7.2 - Roteiro para programação	115
7.3 - Programa em Fortran 90	120
7.4 - Programa em Visual Basic 2017	126
7.5 - Programa em Visual C# 2017	133
7.6 - Exemplos para testar os programas	140
 8. DIMENSIONAMENTO À TORÇÃO	143
8.1 - Apresentação do problema	143
8.2 - Roteiro para programação	143
8.3 - Programa em Fortran 90	149
8.4 - Programa em Visual Basic 2017	156
8.5 - Programa em Visual C# 2017	165
8.6 - Exemplos para testar os programas	173
 9. DIMENSIONAMENTO À FLEXO-TRAÇÃO NORMAL	175
9.1 - Apresentação do problema	175
9.2 - Roteiro para programação	175
9.3 - Programa em Fortran 90	180
9.4 - Programa em Visual Basic 2017	185
9.5 - Programa em Visual C# 2017	191
9.6 - Exemplos para testar os programas	198

Capítulo 1

INTRODUÇÃO

1.1 – Organização do livro

Este livro é um complemento da obra Curso de Concreto Armado¹. Nessa coleção de quatro volumes, foram apresentados, dentre tantos assuntos, os procedimentos de dimensionamento e verificação de seções de concreto armado. Geralmente, ao final de cada capítulo de dimensionamento, foi apresentado um roteiro para organizar o trabalho de cálculo realizado manualmente. Esses roteiros já são, praticamente, roteiros de programação e têm sido utilizados por muitos leitores para a elaboração dos seus próprios programas.

Isso ocorre, por exemplo, nos capítulos sobre dimensionamento à flexão simples (Capítulo 4 e Capítulo 5 do Volume 1), dimensionamento ao esforço cortante (Capítulo 7 do Volume 1), dimensionamento à torção (Capítulo 1 do Volume 4) e dimensionamento à flexo-tração (Capítulo 2 do Volume 4), onde são fornecidos roteiros para um cálculo passo a passo.

Em assuntos mais extensos, como os problemas relacionados à flexo-compressão, foram dadas orientações para a elaboração do programa computacional, como se pode observar no Capítulo 2 do Volume 3.

Enfim, a obra Curso de Concreto Armado, além da preocupação com o cálculo manual, também tem a intenção de despertar nos seus leitores o interesse pela programação. Esse livro vem reforçar esse objetivo.

O livro faz referência explícita aos capítulos e aos volumes de Curso de Concreto Armado. O leitor deve recorrer a essa obra para revisar a teoria. Na tabela 1.1.1, apresentam-se os temas abordados, bem como os capítulos e os volumes onde se encontram os seus desenvolvimentos teóricos.

Tabela 1.1.1 - Temas abordados neste livro

Capítulo deste livro	Assunto	Capítulo onde a teoria é desenvolvida em Curso de Concreto Armado
Capítulo 2	Dimensionamento de seções retangulares à flexão normal simples	Capítulo 4, Volume 1
Capítulo 3	Dimensionamento de seções T à flexão normal simples	Capítulo 5, Volume 1
Capítulo 4	Verificação de seções retangulares submetidas à flexão normal simples	Capítulo 6, Volume 1
Capítulo 5	Dimensionamento ao esforço cortante	Capítulo 7, Volume 1
Capítulo 6	Dimensionamento de seções retangulares à flexo-compressão normal	Capítulo 2, Volume 3
Capítulo 7	Verificação de seções retangulares submetidas à flexo-compressão normal	Capítulos 2 e 3, Volume 3
Capítulo 8	Dimensionamento à torção	Capítulo 1, Volume 4
Capítulo 9	Dimensionamento à flexo-tração normal	Capítulo 2, Volume 4

¹ Araújo, J. M. Curso de Concreto Armado, 4v. 4ª ed., Editora Dunas, Rio Grande, 2014.

1.2 - As linguagens de programação utilizadas

Engenheiros, matemáticos e outros profissionais das áreas de ciências exatas se deparam diariamente com problemas numéricos complexos, que exigem grande esforço de cálculo. O objetivo desses profissionais é a resolução de problemas reais, visando a uma aplicação prática, ou ao desenvolvimento de novas metodologias de cálculo. Nesse sentido, o computador se torna uma ferramenta imprescindível. Para esses profissionais, o computador é um aliado essencial, que resolve a parte repetitiva dos cálculos, deixando que os mesmos reservem suas energias para as atividades intelectuais requeridas por suas profissões. Em geral, eles usam o computador como uma calculadora poderosa, para realizar cálculos pesados, que muitas vezes exigem o emprego de técnicas iterativas e incrementais. Com isso em mente, eles procuram uma linguagem de programação que seja, ao mesmo tempo, simples, precisa, e forneça os resultados desejados de maneira rápida. Eles não estão interessados na forma, na aparência ou na interatividade do software. O software é desenvolvido por eles mesmos, que são os seus próprios usuários. Ou seja, o programa será utilizado pelo programador, que conhece em detalhes o software que ele próprio desenvolveu. O Fortran é uma linguagem muito apropriada para esse fim e vem sendo usado amplamente, desde o seu surgimento em meados do século XX. Os engenheiros que desenvolvem trabalhos de pesquisa em análise numérica, mecânica estrutural, mecânica dos fluidos, dentre outras áreas, utilizam com frequência o Fortran, devido às considerações acima.

Por outro lado, os programadores de computador desenvolvem softwares para usuários que não são eles próprios. Desse modo, os programadores têm uma grande preocupação com o comportamento e com a aceitação dos usuários. O software deve ser visualmente atrativo, deve prover uma boa interação com o usuário e, de preferência, deve fornecer uma saída gráfica. O programador também se preocupa com o comportamento do usuário, prevendo que o mesmo possa introduzir dados incoerentes, não seguir a sequência de cálculo correta para a resolução do problema, etc. Por isso, o programador introduz inúmeras linhas de código com mensagens para o usuário, visando preservar a qualidade do processamento. O Visual Basic e o Visual C# são linguagens de programação adequadas para esse fim.

Para atender a esses dois tipos de necessidades, os programas são apresentados nas seguintes linguagens: Fortran 90, Visual Basic 2017 e Visual C# 2017. O leitor que não utiliza nenhuma dessas linguagens de programação não encontrará dificuldades em adaptar os programas para sua linguagem preferida. Para isto, basta seguir o roteiro de programação que é apresentado no início de cada capítulo.

A escolha das linguagens de programação acima referidas se deu, também, pelo fato de haver versões gratuitas das mesmas. O Fortran pode ser obtido gratuitamente no site da Intel². Ele faz parte do Intel® oneAPI HPC Toolkit. Há, também, o compilador livre do Fortran, conhecido como gfortran³. O Visual Basic 2017 e o Visual C# 2017 fazem parte do Microsoft Visual Studio Community 2017, também disponibilizado gratuitamente no site da Microsoft⁴. No momento dessa reimpressão, a última versão disponibilizada corresponde ao Visual Studio 2019, que pode ser obtido gratuitamente no mesmo link indicado na nota de rodapé.

1.3 – Entrada de dados e separador decimal

Para os programas em Fortran 90, os dados são lidos de um arquivo de texto, o qual pode ser criado com qualquer editor de texto, como o Bloco de Notas do Windows. Nas listagens apresentadas, utiliza-se o arquivo denominado de DADOS.TXT. Os dados decimais devem ser fornecidos utilizando-se o ponto como separador decimal. Por exemplo, se um determinado dado corresponde ao valor real 1/4, o dado deve constar no arquivo DADOS.TXT como 0.25 e não 0,25.

² <https://software.intel.com/pt-br/qualify-for-free-software>

³ <http://mingw-w64.org/doku.php>

⁴ <https://www.visualstudio.com/pt-br/downloads/>

Os programas desenvolvidos em Visual Basic 2017 e Visual C# 2017 utilizam os controles comuns da Caixa de Ferramentas. A maioria dos dados é fornecida em caixas de texto (o controle TextBox). Os programas para verificação de seções submetidas à flexão simples normal, para dimensionamento e verificação de seções submetidas à flexo-compressão normal utilizam, também, uma entrada de dados através de uma grade (o controle DataGridView). Os dados podem ser digitados empregando-se o ponto ou a vírgula como separador decimal. Os programas contêm linhas de código para que o resultado seja independente das configurações de usuário do Windows. Os resultados de saída serão apresentados com ponto ou vírgula, dependendo das configurações de formato do Windows definidas para o computador.

Todos os dados devem ser fornecidos. Se um dado for nulo, deve-se fornecer o valor 0 (zero). Por exemplo, no programa de dimensionamento à torção há três esforços solicitantes de entrada: momento fletor de serviço M_k , esforço cortante de serviço V_k e momento torçor de serviço T_k . Esse programa pode ser utilizado para o dimensionamento à flexão simples, na ausência de esforço cortante e torção. Neste caso, deve-se entrar com 0 (zero) para V_k e 0 (zero) para T_k .

1.4 – Preparando o ambiente de trabalho

Os programas em Fortran 90 são escritos com auxílio de um editor de texto, como o Bloco de Notas do Windows. Ao final da edição, resultará um arquivo de texto que deve ser salvo com a extensão “.f90”. Por exemplo, pode-se salvar o arquivo com o nome de Teste.f90. Esse arquivo contém todas as linhas do código em Fortran 90. As listagens com os códigos dos problemas estudados são apresentadas ao longo deste livro. O leitor pode fazer download desses arquivos no link abaixo:

www.editoradunas.com.br/programas.htm

Após a edição do código-fonte, ele precisa ser compilado para se obter o arquivo executável. Isto pode ser feito de maneiras diferentes, dependendo da versão do Fortran que o usuário possui instalado no computador. Geralmente, utiliza-se a linha de comando para a compilação e para a execução de programas em Fortran. Para isto, é necessário executar o Prompt de Comando e localizar a pasta onde está o arquivo fonte.

Admite-se, por exemplo, que o arquivo Teste.f90 esteja na pasta C:\Flexão Simples e que você tenha instalado o Fortan da Intel. Você deve acessar essa pasta através da linha de comando e escrever “ifort Teste.f90”. A linha completa ficaria assim:

```
C:\Flexão Simples\ifort Teste.f90
```

Ao apertar a tecla Enter, o código-fonte será compilado, dando origem ao arquivo “Teste.exe”. Esse é o arquivo executável do programa Teste.

Se você instalou o compilador livre gfortran, a pasta MinGW-W64 será criada no menu Iniciar. Dentro desta pasta tem o atalho Run Terminal. Esse atalho permite acessar o compilador gfortran. Ao clicar no atalho Run Terminal, abre-se um Prompt de Comando que permite acessar a pasta dos seus arquivos em fortran.f90, fazer a compilação e executar o programa. Supondo que o programa Teste.f90 esteja na pasta C:\Flexão Simples, a linha completa para compilação seria assim:

```
C:\Flexão Simples\gfortran Teste.f90 -o Teste.exe
```

OBS: A parte : “-o” deve estar em letra minúscula. No restante, podem-se usar letras minúsculas e/ou maiúsculas.

Ao apertar a tecla Enter, o código-fonte será compilado, dando origem ao arquivo “Teste.exe”. Esse é o arquivo executável do programa Teste.

Para trabalhar com o Visual Basic 2017 e Visual C# 2017, deve-se executar o Visual Studio 2017, cujo ícone é criado na área de trabalho durante a instalação. Na barra de ferramentas, clique em Arquivo, Novo, Projeto. Na fig. 1.4.1, apresenta-se a janela que se abre.

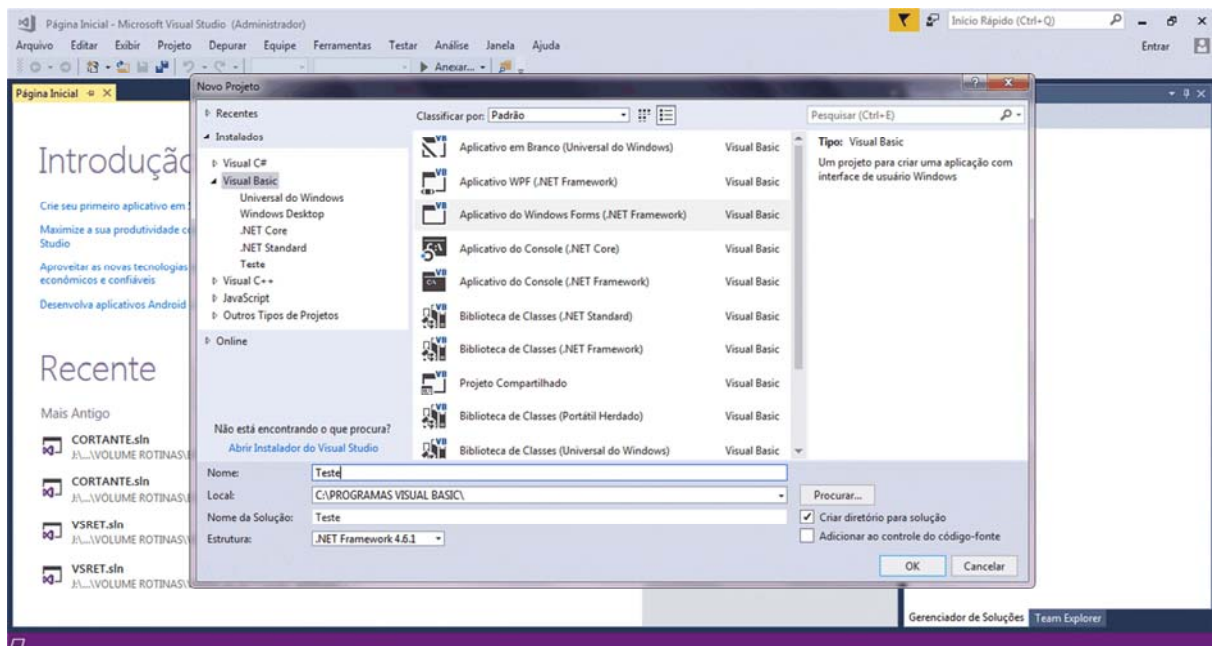


Fig. 1.4.1 – Iniciando novo projeto no Visual Studio 2017

À esquerda da janela Novo Projeto, observam-se os componentes instalados: Visual C#, Visual Basic, Visual C++, JavaScript. Para os desenvolvimentos apresentados neste livro, verifique se os dois primeiros aparecem nesta lista. Caso contrário, reinstale o Visual Studio adicionando o Visual C# e o Visual Basic.

Para criar um novo projeto em Visual Basic, basta selecionar Visual Basic e Aplicativo do Windows Forms (.NET Framework). À direita e em baixo na janela, clique no botão Procurar para localizar a pasta onde o projeto será armazenado e digite o nome para o projeto. Na fig. 1.4.1, o projeto de nome Teste será armazenado na pasta C:\PROGRAMAS VISUAL BASIC.

Após clicar no botão OK, será criado o projeto Teste e o Form1 será exibido na tela. Em cima deste Form1 serão inseridos os diversos componentes do programa. Na barra de ferramentas, clique em Exibir, Caixa de Ferramentas. A fig. 1.4.2 mostra a tela do computador.

Na fig. 1.4.3 mostram-se os Controles Comuns, Contêineres e Dados da Caixa de Ferramentas. Apenas alguns desses controles serão inseridos no Form1.

Para criar um novo projeto em Visual C#, o procedimento é inteiramente análogo.

Ao longo deste livro, são apresentados os Forms e todos os controles utilizados, bem como o código-fonte de cada programa em Visual Basic 2017 e Visual C# 2017. Esses códigos podem ser obtidos no link: www.editoradunas.com.br/programas.htm. O leitor terá apenas o trabalho de criar o Form1 de cada programa e colar o código-fonte. Para isto, basta clicar em Exibir Código na barra de ferramentas e colar o código baixado do link acima.

Observação: A janela de inicialização do Visual Studio 2019 difere da fig. 1.4.1, porém a sequência é a mesma: escolher a linguagem de programação (Visual Basic ou C#) e Aplicativo do Windows Forms (do .NET). Desse modo, os programas apresentados neste livro podem ser utilizados no Visual Studio 2019 sem alteração.

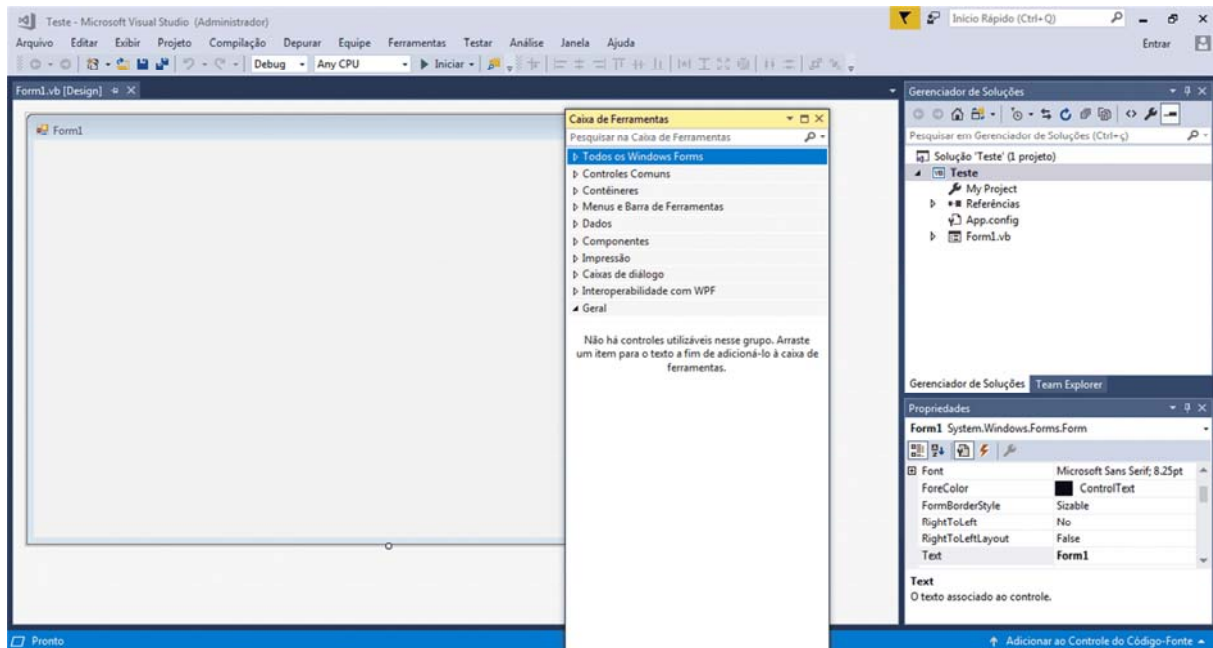


Fig. 1.4.2 – Janela com o Form1 e a Caixa de Ferramentas

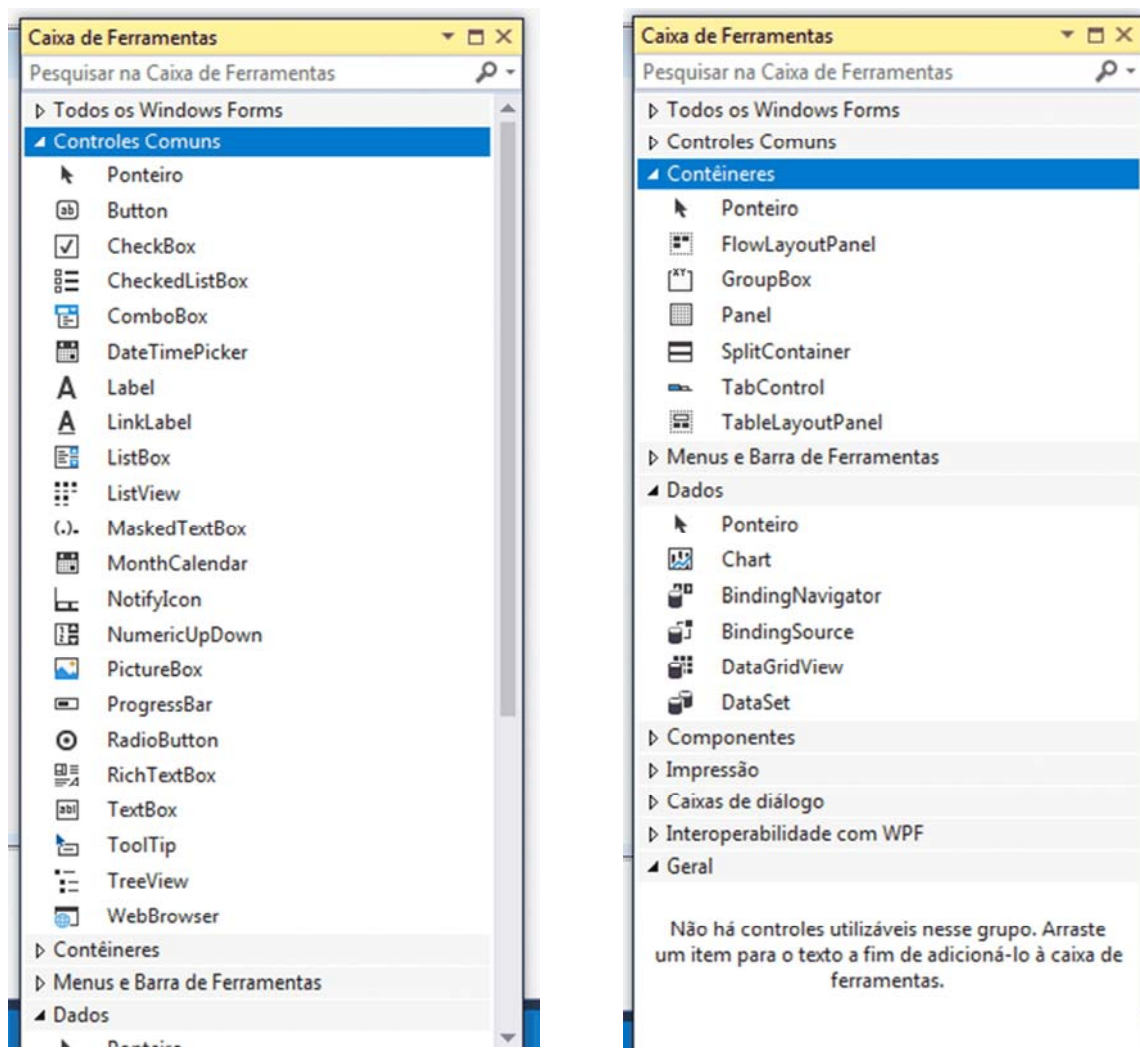


Fig. 1.4.3 – Controles da Caixa de Ferramentas

1.5 - Melhoramentos dos programas

Os programas apresentados resolvem os problemas de dimensionamento e verificação propostos. Eles foram testados e estão prontos para o uso em projetos e/ou na solução de exercícios acadêmicos. Entretanto, não houve preocupação com o acabamento final dos mesmos, particularmente dos programas escritos em Visual Basic e Visual C#. Muitos melhoramentos podem ser feitos e são deixados para o leitor.

O leitor pode implementar testes para controle da entrada de dados e evitar que o usuário forneça dados inconsistentes. Como exemplo, pode-se ler o valor da resistência característica à compressão do concreto, f_{ck} , e verificar se ela se encontra dentro dos limites definidos pela norma NBR-6118⁵. Se $f_{ck} < 20$ MPa ou $f_{ck} > 90$ MPa, o programa apresentará uma mensagem e irá parar o processamento. O mesmo pode ser feito com todos os dados de entrada.

Uma mensagem também pode ser apresentada, se um dado fornecido for nulo ou se a caixa de texto estiver vazia. Em algumas situações, o dado de entrada pode ser nulo, quando se trata de um esforço solicitante. Porém, se o usuário fornecer uma dimensão nula para a seção transversal, haverá um erro no processamento.

Algumas linhas de código podem ser acrescidas para centralizar o formulário na tela do computador. Outro melhoramento consiste em fazer o programa desenhar a seção transversal, escolher o diâmetro das barras da armadura, etc.

Enfim, os programas estão disponibilizados em sua forma bruta. A sua lapidação é deixada como um exercício para o leitor.

1.6 - O software PACON

O software PACON⁶: Programa Auxiliar para projeto de estruturas de CONcreto, desenvolvido pelo Autor, foi escrito em Visual Basic 6.0 e contém todos os programas apresentados neste livro. Esses programas são sub-rotinas que fazem parte de alguns módulos do PACON.

O PACON contém todos os melhoramentos citados anteriormente e resolve esses problemas em um único software. Além disso, o PACON resolve diversos outros problemas como: cálculo de esforços em lajes retangulares, lajes circulares, lajes poligonais e lajes lisas apoiadas em vigas de borda; verificação do cisalhamento em lajes; cálculo de comprimentos de ancoragem e de emendas das armaduras; cálculo de esforços em vigas contínuas; verificação dos estados limites de utilização; determinação das situações de cálculo dos pilares; dimensionamento e verificação de seções vazadas e seções circulares à flexo-compressão normal; dimensionamento e verificação de seções poligonais arbitrárias submetidas à flexo-compressão oblíqua; determinação da rigidez equivalente de pórticos; cálculo de esforços devidos à ação do vento em pórticos planos; cálculo de escadas autoportantes; dentre outros.

Desse modo, o PACON permite a realização de todas as etapas de cálculo do projeto estrutural de um edifício. O cálculo é feito passo a passo, resolvendo-se problema por problema, como apresentado no livro Projeto Estrutural de Edifícios de Concreto Armado⁷.

Nas figuras 1.6.1 a 1.6.3, apresentam-se algumas janelas do PACON.

⁵ ABNT. NBR-6118: Projeto de Estruturas de Concreto. Rio de Janeiro, 2014.

⁶ Araújo, J. M. PACON. 2014 <www.editoradunas.com.br>

⁷ Araújo, J. M. Projeto Estrutural de Edifícios de Concreto Armado. 3ª ed., Editora Dunas, Rio Grande, 2014.

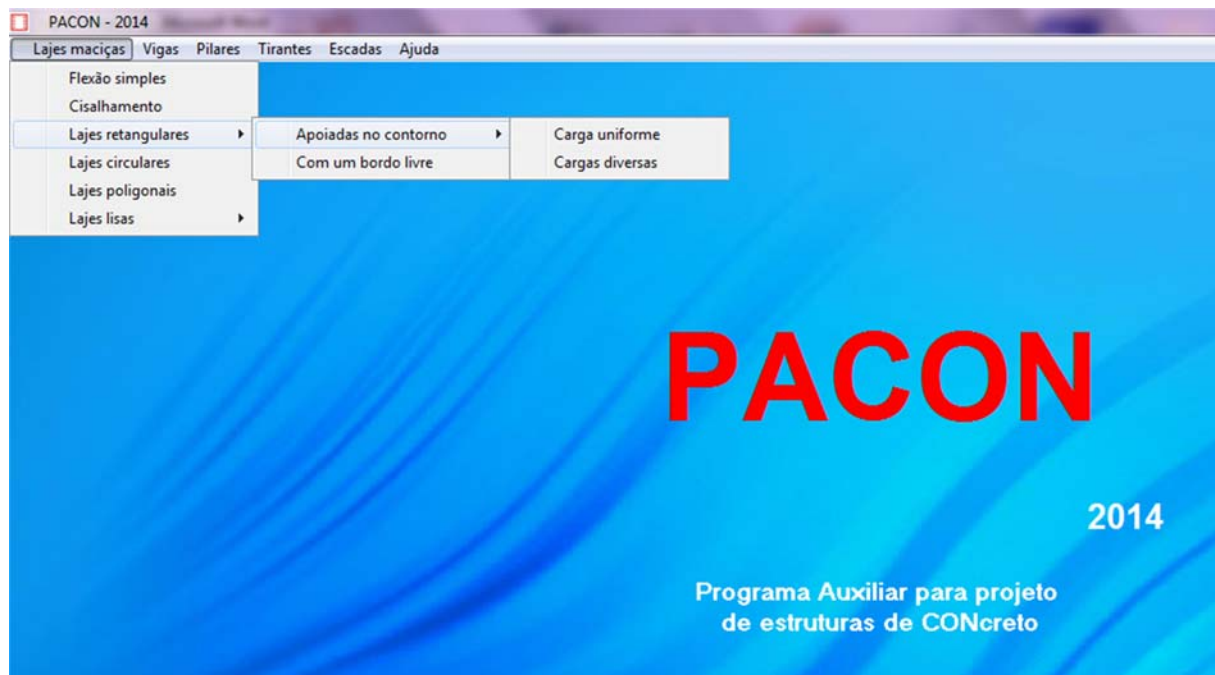


Fig. 1.6.1 - Módulo Lajes do PACON

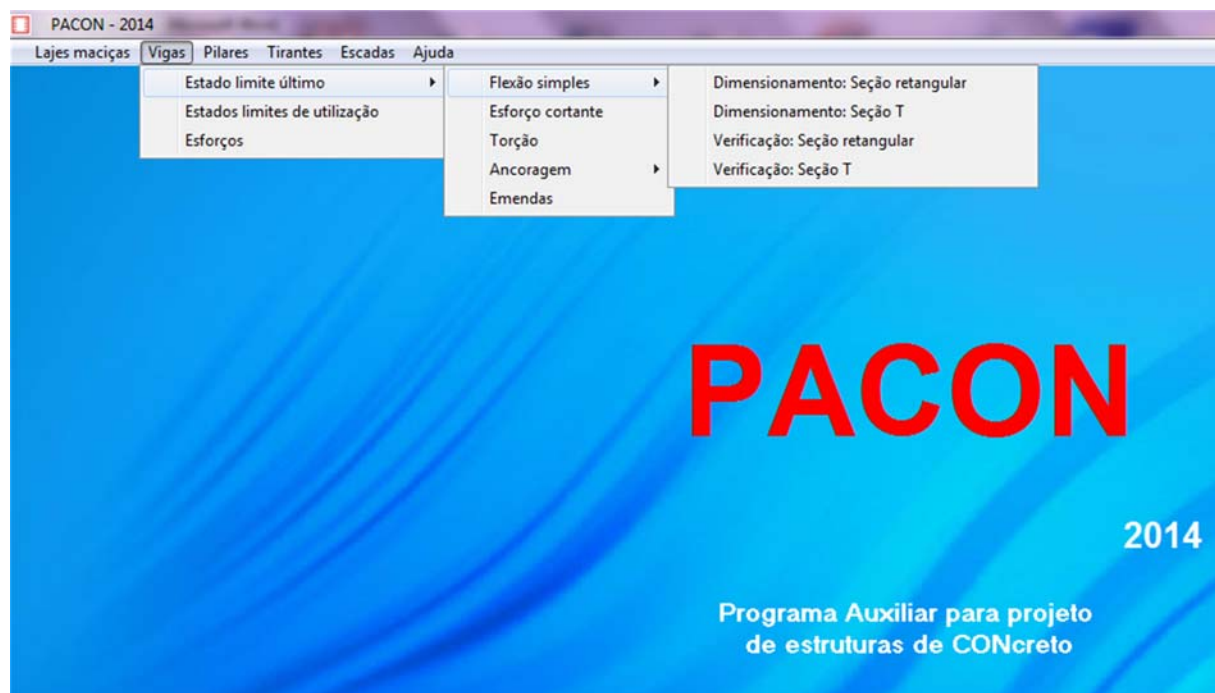


Fig. 1.6.2 - Módulo Vigas do PACON

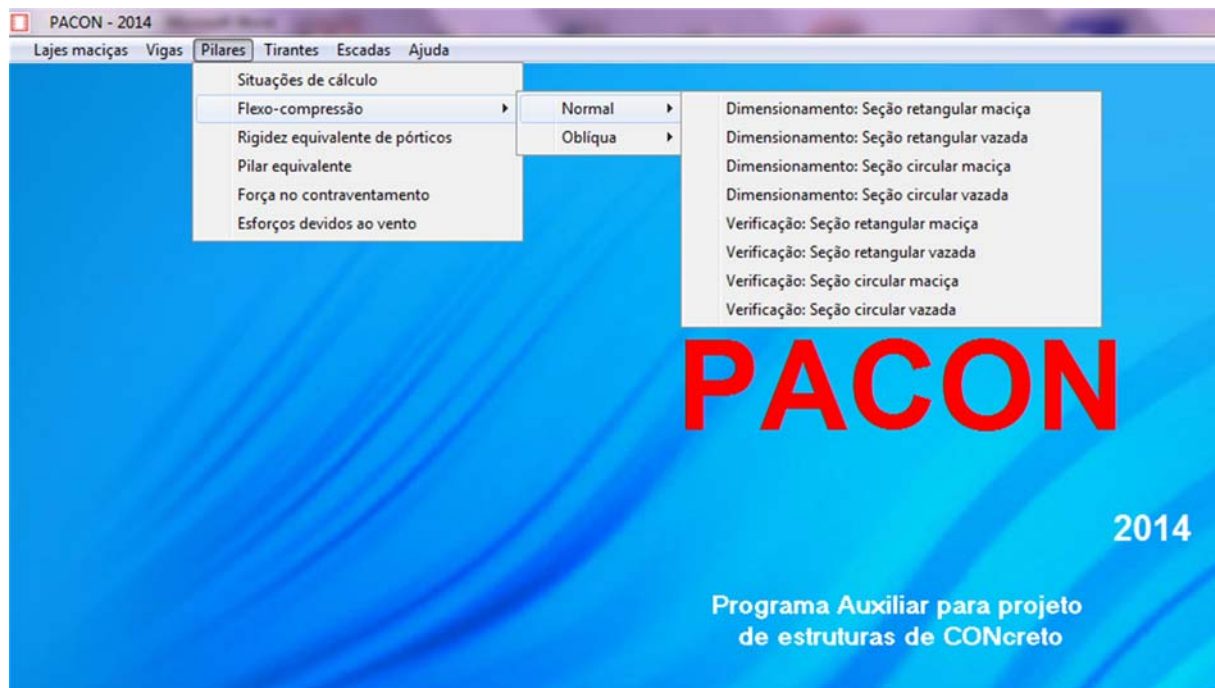


Fig. 1.6.3 - Módulo Pilares do PICON

Capítulo 2

FLEXÃO NORMAL SIMPLES Dimensionamento de Seções Retangulares

2.1 – Apresentação do problema

O objetivo é dimensionar uma seção retangular submetida à flexão simples normal. A geometria da seção é indicada na fig. 2.1.1. A teoria é apresentada no Capítulo 4 do Volume 1.

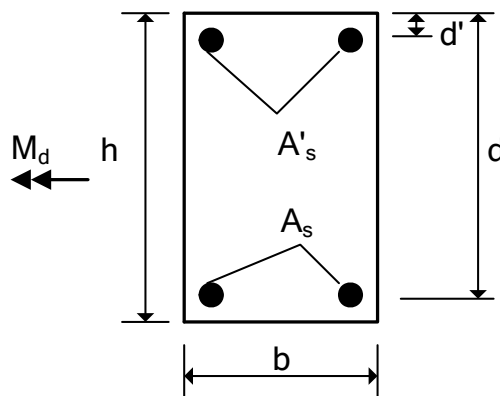


Fig. 2.1.1 - Seção retangular submetida à flexão simples normal

Os dados do problema são os seguintes:

- a) propriedades dos materiais: f_{ck} , f_{yk} , E_s ;
- b) coeficientes parciais de segurança: γ_c , γ_s , γ_f
- c) coeficiente de redistribuição de momentos: $0,75 \leq \beta \leq 1$
- d) dimensões da seção transversal: b , h , d , d' ;
- e) momento fletor de serviço: M_k (valor característico na hipótese de uma análise linear).

Os valores requeridos são as áreas de aço A_s e A'_s .

2.2 – Roteiro para programação

A) Programa principal

1) Entrada de dados

Ler as propriedades dos materiais: f_{ck} (MPa), f_{yk} (MPa), E_s (GPa)

Ler os coeficientes parciais de segurança: γ_c , γ_s , γ_f

Ler o coeficiente de redistribuição de momentos: β

Ler as dimensões da seção (em cm): b , h , d , d'

Ler o momento fletor de serviço em kNm: M_k

Fim da entrada de dados e início dos cálculos.

2) Parâmetros do diagrama retangular para o concreto e profundidade limite da linha neutra

- Se $f_{ck} \leq 50$ MPa:

$$\lambda = 0,8$$

$$\alpha_c = 0,85$$

$$\varepsilon_u \left(\frac{o}{oo} \right) = 3,5$$

$$\xi_{lim} = 0,8\beta - 0,35$$

- Se $f_{ck} > 50$ MPa:

$$\lambda = 0,8 - \frac{(f_{ck} - 50)}{400}$$

$$\alpha_c = 0,85 \left[1 - \frac{(f_{ck} - 50)}{200} \right]$$

$$\varepsilon_u \left(\frac{o}{oo} \right) = 2,6 + 35 \left(\frac{90 - f_{ck}}{100} \right)^4$$

$$\xi_{lim} = 0,8\beta - 0,45$$

3) Conversão de unidades para kN e cm

$$M_k = 100M_k$$

$$f_{ck} = f_{ck} / 10$$

$$f_{yk} = f_{yk} / 10$$

$$E_s = 100E_s$$

4) Resistências e momento de cálculo

$$f_{cd} = f_{ck} / \gamma_c$$

$$\sigma_{cd} = \alpha_c f_{cd}$$

$$f_{yd} = f_{yk} / \gamma_s$$

$$M_d = \gamma_f M_k$$

5) Parâmetro geométrico

$$\delta = d' / d$$

6) Momento limite

$$\mu_{lim} = \lambda \xi_{lim} (1 - 0,5 \lambda \xi_{lim})$$

7) Momento reduzido solicitante

$$\mu = \frac{M_d}{bd^2\sigma_{cd}}$$

8) Solução com armadura simples

- Se $\mu \leq \mu_{\lim}$:

$$\xi = \frac{1 - \sqrt{1 - 2\mu}}{\lambda}$$

$$A_s = \lambda \xi bd \frac{\sigma_{cd}}{f_{yd}}$$

$$A'_s = 0.$$

9) Solução com armadura dupla

- Se $\mu > \mu_{\lim}$:

- ✓ Evitar o uso de armadura dupla no domínio 2:

$$\xi_a = \frac{\varepsilon_u}{\varepsilon_u + 10\%_{00}}$$

Se $\xi_{\lim} < \xi_a$, colocar mensagem para o usuário aumentar as dimensões da seção transversal e parar o processamento.

- ✓ Evitar o caso em que $\xi_{\lim} \leq \delta$, pois a armadura de compressão ficaria tracionada. Colocar mensagem para o usuário aumentar as dimensões da seção transversal e parar o processamento.

Tensão na armadura de compressão:

$$\varepsilon'_s = \varepsilon_u \left(\frac{\xi_{\lim} - \delta}{\xi_{\lim}} \right)$$

Chamar uma sub-rotina para calcular a tensão σ'_{sd} .

$$A'_s = \frac{(\mu - \mu_{\lim})bd\sigma_{cd}}{(1 - \delta)\sigma'_{sd}}$$

$$A_s = \left(\lambda \xi_{\lim} + \frac{\mu - \mu_{\lim}}{1 - \delta} \right) \frac{bd\sigma_{cd}}{f_{yd}}$$

10) Armadura mínima

Voltar f_{ck} e f_{yd} para MPa:

$$f_{ck} = 10f_{ck}$$

$$f_{yd} = 10 f_{yd}$$

- Se $f_{ck} \leq 50 \text{ MPa}$:

$$\rho_{\min} = \frac{0,078 f_{ck}^{2/3}}{f_{yd}}$$

- Se $f_{ck} > 50 \text{ MPa}$:

$$\rho_{\min} = \frac{0,5512 \ln(1 + 0,11 f_{ck})}{f_{yd}}$$

Se $\rho_{\min} < 0,15\% \rightarrow$ fazer $\rho_{\min} = 0,15\%$

$$A_{s,\min} = \rho_{\min} b h$$

Se $A_s < A_{s,\min}$, adotar $A_s = A_{s,\min}$

Imprimir ou mostrar o resultado: A_s e A'_s (em cm^2)

Fim do programa principal.

B) Sub-rotina tensão

Parâmetros de entrada: E_s , ε_s , f_{yd}

Parâmetro de saída: σ_{sd}

1) Trabalhar com o valor absoluto da deformação

$$\varepsilon_{ss} = |\varepsilon_s|$$

2) Deformação de escoamento de cálculo do aço

$$\varepsilon_{yd} = f_{yd} / E_s$$

3) Cálculo da tensão

- Se $\varepsilon_{ss} < \varepsilon_{yd}$:

$$\sigma_{sd} = E_s \varepsilon_{ss}$$

- Se $\varepsilon_{ss} \geq \varepsilon_{yd}$:

$$\sigma_{sd} = f_{yd}$$

4) Acertando o sinal da tensão

Se $\varepsilon_s < 0$, fazer $\sigma_{sd} = -\sigma_{sd}$

Fim da sub-rotina.

2.3 – Programa em Fortran 90

A listagem a seguir apresenta o código do programa em Fortran 90. Os dados são lidos do arquivo DADOS.TXT. Para facilitar a comparação com os programas em Visual Basic e Visual C#, cada dado é lido em uma linha do arquivo. As mensagens de alerta e os resultados são mostrados na tela. Os dados devem ser fornecidos usando somente o ponto como separador decimal. O arquivo com a listagem do código-fonte em Fortran 90 pode ser obtido em www.editoradunas.com.br/programas.htm.

LISTAGEM DO CÓDIGO-FONTE EM FORTRAN 90

```
!  
!Dimensionamento de seções retangulares à flexão normal simples  
!  
!PROGRAMA PRINCIPAL  
!  
!Declaração variáveis reais de dupla precisão  
IMPLICIT REAL*8(A-H,O-Z)  
!  
!Abertura do arquivo de dados  
OPEN(1,FILE='DADOS.TXT')  
!  
!LEITURA DOS DADOS  
!  
!Os dados são lidos do arquivo DADOS.TXT  
!Cada dado deve estar em uma linha do arquivo  
!  
!Resistência característica à compressão do concreto em MPa  
read(1,*)fck  
!  
!Tensão de escoamento característica do aço em MPa  
read(1,*)fyk  
!  
!Módulo de elasticidade do aço em GPa  
read(1,*)es  
!  
!Coeficientes parciais de segurança:  
!para o concreto  
read(1,*)gamac  
!  
!para o aço  
read(1,*)gamas  
!  
!para o momento fletor  
read(1,*)gamaf  
!  
!Coeficiente beta de redistribuição de momentos  
read(1,*)bduct  
!  
!Largura da seção transversal  
read(1,*)b  
!  
!Altura da seção transversal
```

```

read(1,*)h
!
!Altura útil
read(1,*)d
!
!Parâmetro d!
read(1,*)dl
!
!Momento fletor de serviço em kNm
read(1,*)amk
!
!
!FIM DA ENTRADA DE DADOS
!
!INÍCIO DOS CÁLCULOS
!
!
!Parâmetros do diagrama retangular
If(fck<=50)Then
    alamb=0.8
    alfac=0.85
    eu=3.5
    qlim=0.8*bduct-0.35
Else
    alamb=0.8-(fck-50)/400
    alfac=0.85*(1-(fck-50)/200)
    eu=2.6+35*((90-fck)/100)**4
    qlim=0.8*bduct-0.45
End If
!
!Conversão de unidades: transformando para kN e cm
amk=100*amk
fck=fck/10
fyk=fyk/10
es=100*es
!
!Resistências de cálculo
fcd=fck/gamac
tcd=alfac*fcd
fyd=fyk/gamas
amd=gamaf*amk
!
!Parâmetro geométrico
delta=dl/d
!
!Momento limite
amilim=alamb*qlim*(1-0.5*alamb*qlim)
!
!Momento reduzido solicitante
ami=amd/(b*d*d*tcd)
!
If(ami<=amilim)Then
!Armadura simples
    qsi=(1-sqrt(1-2*ami))/alamb
    aas=alamb*qsi*b*d*tcd/fyd
    asl=0
Else

```

```

!Armadura dupla
!
! Evitando armadura dupla no domínio 2
  qsia=eu/(eu+10)
  If(qlim<qsia)Then
!
! Está resultando armadura dupla no domínio 2.
!   Colocar mensagem para o usuário aumentar as dimensões da seção transversal e parar o
processamento
    write(*,10)
10  format(//,5x,'Resultou armadura dupla no dominio 2. Aumente as dimensoes da secao transversal')
    goto 100
  End If
!
! Eliminando o caso em que qlim<delta
! Se isto ocorrer, a armadura de compressão estará tracionada
!
  If(qlim<=delta)Then
!
!   Colocar mensagem para o usuário aumentar as dimensões da seção transversal e parar o
processamento
!
    write(*,20)
20  format(//,5x,'Aumente as dimensoes da secao transversal')
    goto 100
  End If
!
! Deformação da armadura de compressão
  esl=eu*(qlim-delta)/qlim
  esl=esl/1000
! Tensão na armadura de compressão
! Chamar subrotina
  Call Tensao(es,esl,fyd,tsl)
  asl=(ami-amilim)*b*d*tcd/((1-delta)*tsl)
  aas=(alamb*qlim+(ami-amilim)/(1-delta))*b*d*tcd/fyd
End If
!
!Armadura mínima
a=2./3.
fck=10*fck
fyd=10*fyd
If(fck<=50)Then
  romin=0.078*(fck**a)/fyd
Else
  romin=0.5512*Log(1+0.11*fck)/fyd
End If
If(romin<0.0015) Then
  romin=0.0015
End If
!
asmin=romin*b*h
If(aas<asmin) Then
  aas=asmin
End If
!
!
!MOSTRAR O RESULTADO NA TELA

```



```

!Área da armadura tracionada: aas
!Área da armadura comprimida: asl
!
write(*,30)aas
30 format(/,5x,'Area da armadura tracionada=',f6.2)
write(*,40)asl
40 format(/,5x,'Area da armadura comprimida=',f6.2)
100 aux=0 !terminando o programa
End
!
!
SUBROUTINE Tensao(es,esl,fyd,tsl)
IMPLICIT REAL*8(A-H,O-Z)
!
!Calcula a tensão no aço
!es = módulo de elasticidade do aço em kN/cm2
!esl = deformação de entrada em por mil
!fyd = tensão de escoamento de cálculo em kN/cm2
!tsl = tensão de saída em kN/cm2
!
!Trabalhando com deformação positiva
ess=Abs(esl)
eyd=fyd/es
If(ess<eyd)Then
    tsl=es*ess
Else
    tsl=fyd
End If
!Trocando o sinal se necessário
If(esl<0)Then
    tsl=-tsl
End If
Return
End

```

2.4 – Programa em Visual Basic 2017

Na fig. 2.4.1 apresenta-se o formulário do programa escrito em Visual Basic. Os dados podem ser fornecidos usando o ponto ou a vírgula como separador decimal.

Para criar o projeto, procede-se como descrito na seção 1.4 do capítulo 1. Utilizando a Caixa de Ferramentas, são introduzidos os controles no Form1. A propriedade Text do Form1 é “Dimensionamento à Flexão Simples Normal – Seção retangular”, como aparece no topo do formulário.

O formulário contém os seguintes controles:

Tipo	Nome	Descrição
PictureBox	PictureBox1	Imagem da seção
Label	Label1	fck (MPa) =
TextBox	TextBox1	Valor de fck
Label	Label2	fyk (MPa) =
TextBox	TextBox2	Valor de fyk
Label	Label3	Es (GPa) =
TextBox	TextBox3	Valor de Es
Label	Label4	Gamac =
TextBox	TextBox4	Valor de Gamac
Label	Label5	Gamas =

TextBox	TextBox5	Valor de Gamaf
Label	Label6	Gamaf =
TextBox	TextBox6	Valor de Gamaf
Label	Label7	Beta =
TextBox	TextBox7	Valor de Beta
Label	Label8	b (cm) =
TextBox	TextBox8	Valor de b
Label	Label9	h (cm) =
TextBox	TextBox9	Valor de h
Label	Label10	d (cm) =
TextBox	TextBox10	Valor de d
Label	Label11	d' (cm) =
TextBox	TextBox11	Valor de d'
Label	Label12	Mk (kNm) =
TextBox	TextBox12	Valor de Mk
Button	Button1	Calcular
Label	Label13	As (cm2) =
TextBox	TextBox13	Valor de As
Label	Label14	A's (cm2) =
TextBox	TextBox14	Valor de A's
GroupBox	GroupBox1	Materiais
GroupBox	GroupBox2	Coeficientes parciais
GroupBox	GroupBox3	Coeficiente de redistribuição
GroupBox	GroupBox4	Dimensões da seção
GroupBox	GroupBox5	Momento fletor de serviço
GroupBox	GroupBox6	RESPOSTA

Dimensionamento à Flexão Simples Normal - Seção retangular

Materiais
 f_{ck} (MPa) = f_{yk} (MPa) = E_s (GPa) =

Coeficientes parciais
 γ_{mac} = γ_{mas} = γ_{maf} =

Coeficiente de redistribuição
 β =

Dimensões da seção
 b (cm) = h (cm) =
 d (cm) = d' (cm) =

Momento fletor de serviço
 M_k (kNm) =

RESPOSTA
 A_s (cm²) = A'_s (cm²) =

Fig. 2.4.1 – Formulário do programa em Visual Basic 2017

O arquivo com a listagem do código-fonte em Visual Basic 2017 pode ser obtido em www.editoradunas.com.br/programas.htm. Após a elaboração do formulário, basta clicar com o botão direito do mouse sobre o formulário para Exibir Código e colar o código-fonte.

LISTAGEM DO CÓDIGO-FONTE EM VISUAL BASIC 2017

Public Class Form1

Public ts1 As Double

Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click

Dim fck, fyk, es, gamac, gamas, gamaf, bduct As Double

Dim b, d, h, dl, delta, amk As Double

Dim alamb, alfac, eu, qlim, ami, amilim As Double

Dim fcd, tcd, fyd, amd As Double

Dim qsi, aas, asl, qsia, esl, romin, asmin, a As Double

,

,

'Esta subrotina corresponde ao botão CALCULAR.

,

,

,

'Dimensionamento de seções retangulares à flexão normal simples

,

'ENTRADA DE DADOS

,

'Os dados são lidos das caixas de texto do formulário

,

'Resistência característica à compressão do concreto em MPa

'Substituindo a vírgula por ponto

TextBox1.Text = Replace(TextBox1.Text, ",", ".")

fck = Val(TextBox1.Text)

,

'Tensão de escoamento característica do aço em MPa

TextBox2.Text = Replace(TextBox2.Text, ",", ".")

fyk = Val(TextBox2.Text)

,

'Módulo de elasticidade do aço em GPa

TextBox3.Text = Replace(TextBox3.Text, ",", ".")

es = Val(TextBox3.Text)

,

'Coeficientes parciais de segurança:

'para o concreto

TextBox4.Text = Replace(TextBox4.Text, ",", ".")

gamac = Val(TextBox4.Text)

,

'para o aço

TextBox5.Text = Replace(TextBox5.Text, ",", ".")

gamas = Val(TextBox5.Text)

,

'para o momento fletor

TextBox6.Text = Replace(TextBox6.Text, ",", ".")

gamaf = Val(TextBox6.Text)

,

'Coeficiente beta de redistribuição de momentos

TextBox7.Text = Replace(TextBox7.Text, ",", ".")

bduct = Val(TextBox7.Text)

,

```

'Largura da seção transversal
TextBox8.Text = Replace(TextBox8.Text, ",", ".")
b = Val(TextBox8.Text)
,

'Altura da seção transversal
TextBox9.Text = Replace(TextBox9.Text, ",", ".")
h = Val(TextBox9.Text)
,

'Altura útil
TextBox10.Text = Replace(TextBox10.Text, ",", ".")
d = Val(TextBox10.Text)
,

'Parâmetro d'
TextBox11.Text = Replace(TextBox11.Text, ",", ".")
dl = Val(TextBox11.Text)
,

'Momento fletor de serviço em kNm
TextBox12.Text = Replace(TextBox12.Text, ",", ".")
amk = Val(TextBox12.Text)
,

'FIM DA ENTRADA DE DADOS
,

'INÍCIO DOS CÁLCULOS
,
,

'Parâmetros do diagrama retangular
If (fck <= 50) Then
    alamb = 0.8
    alfac = 0.85
    eu = 3.5
    qlim = 0.8 * bduct - 0.35
Else
    alamb = 0.8 - (fck - 50) / 400
    alfac = 0.85 * (1 - (fck - 50) / 200)
    eu = 2.6 + 35 * ((90 - fck) / 100) ^ 4
    qlim = 0.8 * bduct - 0.45
End If
,

'Conversão de unidades: transformando para kN e cm
amk = 100 * amk
fck = fck / 10
fyk = fyk / 10
es = 100 * es
,

'Resistências de cálculo
fcd = fck / gamac
tcd = alfac * fcd
fyd = fyk / gamas
amd = gamaf * amk
,

'Parâmetro geométrico
delta = dl / d
,

'Momento limite
amilim = alamb * qlim * (1 - 0.5 * alamb * qlim)
,

'Momento reduzido solicitante

```

```

ami = amd / (b * d * d * tcd)
,
If (ami <= amilim) Then
    'Armadura simples
    qsi = (1 - Math.Sqrt(1 - 2 * ami)) / alamb
    aas = alamb * qsi * b * d * tcd / fyd
    asl = 0
Else
    'Armadura dupla
    ,
    'Evitando armadura dupla no domínio 2
    qsia = eu / (eu + 10)
    If (qlim < qsia) Then
        ,
        ' Está resultando armadura dupla no domínio 2.
        ' Colocar mensagem para o usuário aumentar as dimensões da seção transversal e parar o
processamento
        ,
        MsgBox("Resultou armadura dupla no domínio 2. Aumente as dimensões da seção
transversal", vbOKOnly + vbExclamation)
        Exit Sub
    End If
    ,
    ' Eliminando o caso em que qlim<delta
    ' Se isto ocorrer, a armadura de compressão estará tracionada
    ,
    If (qlim <= delta) Then
        ,
        ' Colocar mensagem para o usuário aumentar as dimensões da seção transversal e parar o
processamento
        ,
        MsgBox("Aumente as dimensões da seção transversal", vbOKOnly + vbExclamation)
        Exit Sub
    End If
    ,
    'Deformação da armadura de compressão
    esl = eu * (qlim - delta) / qlim
    esl = esl / 1000
    'Tensão na armadura de compressão
    'Chamar Sub-rotina
    Call Tensao(es, esl, fyd)
    asl = (ami - amilim) * b * d * tcd / ((1 - delta) * tsl)
    aas = (alamb * qlim + (ami - amilim) / (1 - delta)) * b * d * tcd / fyd
End If
,
'Armadura mínima
a = 2 / 3
fck = 10 * fck
fyd = 10 * fyd
If (fck <= 50) Then
    romin = 0.078 * (fck ^ a) / fyd
Else
    romin = 0.5512 * Math.Log(1 + 0.11 * fck) / fyd
End If
If (romin < 0.0015) Then
    romin = 0.0015
End If

```

```

'
asmin = romin * b * h
If (aas < asmin) Then
    aas = asmin
End If
'

'Convertendo a saída para duas casas decimais
aas = FormatNumber(aas, 2)
asl = FormatNumber(asl, 2)
'

'MOSTRAR O RESULTADO
'Área da armadura tracionada: aas
'Área da armadura comprimida: asl
'

TextBox13.Text = aas
TextBox14.Text = asl
End Sub

Private Sub Tensao(es, esl, fyd)
    Dim ess, eyd As Double
    '
    'Calcula a tensão no aço
    'es = módulo de elasticidade do aço em kN/cm2
    'esl = deformação de entrada
    'fyd = tensão de escoamento de cálculo em kN/cm2
    'tsl = tensão de saída em kN/cm2
    '
    'Trabalhando com deformação positiva
    ess = Math.Abs(esl)
    eyd = fyd / es
    If (ess < eyd) Then
        tsl = es * ess
    Else
        tsl = fyd
    End If
    'Trocando o sinal se necessário
    If (esl < 0) Then
        tsl = -tsl
    End If

End Sub

End Class

```

2.5 – Programa em Visual C# 2017

Na fig. 2.5.1 apresenta-se o formulário do programa escrito em Visual C# 2017. O formulário é inteiramente análogo ao do programa em Visual Basic 2017. Os dados podem ser fornecidos usando o ponto ou a vírgula como separador decimal. Os controles são os mesmos do programa em Visual Basic.

O arquivo com a listagem do código-fonte em Visual C# 2017 pode ser obtido em www.editoradunas.com.br/programas.htm. Após a elaboração do formulário, basta clicar com o botão direito do mouse sobre o formulário para Exibir Código e colar o código-fonte.

Dimensionamento à Flexão Simples Normal - Seção Retangular

Diagrama da Seção: Retângulo com largura b , altura total h , e altura útil d . Áreas de armadura: A'_s (topo) e A_s (fundo).

Campos de Entrada:

- Materiais:** f_{ck} (MPa)=, f_{yk} (MPa)=, E_s (GPa)=
- Coefficientes parciais:** γ_{mac} =, γ_{mas} =, γ_{maf} =
- Coefficiente de redistribuição:** β =
- Dimensões da seção:** b (cm)=, h (cm)=, d (cm)=, d' (cm)=
- Momento fletor de serviço:** M_k (kNm)=

Campos de Saída (Resposta): A_s (cm²)=, A'_s (cm²)=

Botão: **CALCULAR**

Fig. 2.5.1 – Formulário do programa em Visual C# 2017

LISTAGEM DO CÓDIGO-FONTE EM VISUAL C# 2017

```
using System;
using System.Globalization;
using System.Windows.Forms;

namespace FSNSRET
{
    public partial class Form1 : Form
    {
        //Declarando variáveis públicas
        public double es, esl, fyd, tsl;
        public double aas, asl;
        public Form1()
        {
            InitializeComponent();
        }

        private void button1_Click(object sender, EventArgs e)
        {
            double fck, fyk, gamac, gamas, gamaf, bduct;
            double b, d, h, dl, delta, amk;
            double alamb, alfac, eu, qlim, ami, amilim;
            double fcd, tcd, amd;
            double qsi, qsia, romin, asmin, a;

            //
            //
        }
    }
}
```

```
//Esta subrotina corresponde ao botão CALCULAR.
//
//
//Dimensionamento de seções retangulares à flexão normal simples
//
//ENTRADA DE DADOS
//
//Os dados são lidos das caixas de texto do formulário
//
//Resistência característica à compressão do concreto em MPa
//Substituindo a vírgula por ponto
textBox1.Text = textBox1.Text.Replace(",", ".");
fck = Convert.ToDouble(textBox1.Text, CultureInfo.InvariantCulture);
//
//Tensão de escoamento característica do aço em MPa
textBox2.Text = textBox2.Text.Replace(",", ".");
fyk = Convert.ToDouble(textBox2.Text, CultureInfo.InvariantCulture);
//
//Módulo de elasticidade do aço em GPa
textBox3.Text = textBox3.Text.Replace(",", ".");
es = Convert.ToDouble(textBox3.Text, CultureInfo.InvariantCulture);
//
//Coeficientes parciais de segurança:
//para o concreto
textBox4.Text = textBox4.Text.Replace(",", ".");
gamac = Convert.ToDouble(textBox4.Text, CultureInfo.InvariantCulture);
//
//para o aço
textBox5.Text = textBox5.Text.Replace(",", ".");
gamas = Convert.ToDouble(textBox5.Text, CultureInfo.InvariantCulture);
//
//para o momento fletor
textBox6.Text = textBox6.Text.Replace(",", ".");
gamaf = Convert.ToDouble(textBox6.Text, CultureInfo.InvariantCulture);
//
//Coeficiente beta de redistribuição de momentos
textBox7.Text = textBox7.Text.Replace(",", ".");
bduct = Convert.ToDouble(textBox7.Text, CultureInfo.InvariantCulture);
//
//Largura da seção transversal
textBox8.Text = textBox8.Text.Replace(",", ".");
b = Convert.ToDouble(textBox8.Text, CultureInfo.InvariantCulture);
//
//Altura da seção transversal
textBox9.Text = textBox9.Text.Replace(",", ".");
h = Convert.ToDouble(textBox9.Text, CultureInfo.InvariantCulture);
//
//Altura útil
textBox10.Text = textBox10.Text.Replace(",", ".");
d = Convert.ToDouble(textBox10.Text, CultureInfo.InvariantCulture);
//
//Parâmetro d'
textBox11.Text = textBox11.Text.Replace(",", ".");
dl = Convert.ToDouble(textBox11.Text, CultureInfo.InvariantCulture);
//
//Momento fletor de serviço em kNm
textBox12.Text = textBox12.Text.Replace(",", ".");
```



```

amk = Convert.ToDouble(textBox12.Text, CultureInfo.InvariantCulture);
//
//
//FIM DA ENTRADA DE DADOS
//
//INÍCIO DOS CÁLCULOS
//
//
//Parâmetros do diagrama retangular
if (fck <= 50)
{
    alamb = 0.8;
    alfac = 0.85;
    eu = 3.5;
    qlim = 0.8 * bduct - 0.35;
}
else
{
    alamb = 0.8 - (fck - 50) / 400;
    alfac = 0.85 * (1 - (fck - 50) / 200);
    a = (90 - fck) / 100;
    eu = 2.6 + 35 * Math.Pow(a, 4);
    qlim = 0.8 * bduct - 0.45;
}
//
//Conversão de unidades: transformando para kN e cm
amk = 100 * amk;
fck = fck / 10;
fyk = fyk / 10;
es = 100 * es;
//
//Resistências de cálculo
fcd = fck / gamac;
tcd = alfac * fcd;
fyd = fyk / gamas;
amd = gamaf * amk;
//
//Parâmetro geométrico
delta = dl / d;
//
//Momento limite
amilim = alamb * qlim * (1 - 0.5 * alamb * qlim);
//
//Momento reduzido solicitante
ami = amd / (b * d * d * tcd);
//
if (ami <= amilim)
{
    //Armadura simples
    qsi = (1 - Math.Sqrt(1 - 2 * ami)) / alamb;
    aas = alamb * qsi * b * d * tcd / fyd;
    asl = 0;
}
if (ami > amilim)
{
    //Armadura dupla
    //

```

```

//Evitando armadura dupla no domínio 2
qsia = eu / (eu + 10);
if (qlim < qsia)
{
    //
    // Está resultando armadura dupla no domínio 2.
    // Colocar mensagem para o usuário aumentar as dimensões da seção transversal e parar o
processamento
    //
    MessageBox.Show("Resultou armadura dupla no domínio 2. Aumente as dimensões da
seção transversal");
    return;
}
//
// Eliminando o caso em que qlim<delta
// Se isto ocorrer, a armadura de compressão estará tracionada
//
if (qlim <= delta)
{
    //
    // Colocar mensagem para o usuário aumentar as dimensões da seção transversal e parar o
processamento
    //
    MessageBox.Show("Aumente as dimensões da seção transversal");
    return;
}
//
//Deformação da armadura de compressão
esl = eu * (qlim - delta) / qlim;
esl = esl / 1000;
// Tensão na armadura de compressão
// Chamar Sub-rotina
Tensao();
asl = (ami - amilim) * b * d * tcd / ((1 - delta) * tsl);
aas = (alamb * qlim + (ami - amilim) / (1 - delta)) * b * d * tcd / fyd;
}
//
//Armadura mínima
a = 2.0 / 3.0;
fck = 10 * fck;
fyd = 10 * fyd;
if (fck <= 50)
{
    romin = 0.078 * Math.Pow(fck, a) / fyd;
}
else
{
    romin = 0.5512 * Math.Log(1 + 0.11 * fck) / fyd;
}
if (romin < 0.0015)
{
    romin = 0.0015;
}
//
asmin = romin * b * h;
if (aas < asmin)
{

```

```

        aas = asmin;
    }
    //
    //Convertendo a saída para duas casas decimais
    //
    decimal saida1 = Decimal.Round(Convert.ToDecimal(aas), 2);
    decimal saida2 = Decimal.Round(Convert.ToDecimal(asl), 2);
    //
    //MOSTRAR O RESULTADO
    //Área da armadura tracionada: aas
    //Área da armadura comprimida: asl
    //
    textBox13.Text = Convert.ToString(saida1);
    textBox14.Text = Convert.ToString(saida2);

}

private void Tensao()
{
    double ess, eyd;
    //
    //Calcula a tensão no aço
    //es = módulo de elasticidade do aço em kN/cm2
    //esl = deformação de entrada
    //fyd = tensão de escoamento de cálculo em kN/cm2
    //tsl = tensão de saída em kN/cm2
    //
    //Trabalhando com deformação positiva
    ess = Math.Abs(esl);
    eyd = fyd / es;
    if(ess < eyd)
    {
        tsl = es * ess;
    }
    else
    {
        tsl = fyd;
    }
    //Trocando o sinal se necessário
    if(esl < 0)
    {
        tsl = -tsl;
    }
}
}
}

```

2.6 – Exemplos para testar os programas

Os resultados apresentados a seguir servem para o leitor testar os programas. Os exemplos correspondem à seção retangular da fig. 2.6.1. Esses exemplos foram resolvidos manualmente no Capítulo 4 do Volume 1.

A tensão de escoamento característica do aço é $f_{yk} = 500$ MPa e o módulo de elasticidade é $E_s = 200$ GPa. Os coeficientes parciais de segurança são $\gamma_c = 1,4$, $\gamma_s = 1,15$

e $\gamma_f = 1,4$. O coeficiente de redistribuição é $\beta = 1$, ou seja, sem redistribuição de momentos.

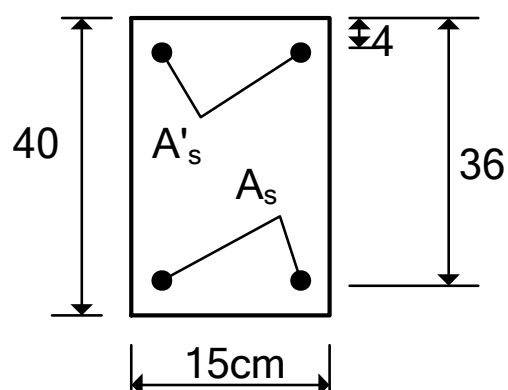


Fig. 2.6.1 - Dimensões da seção retangular

Na tabela 2.6.1 apresentam-se os resultados para quatro exemplos.

Tabela 2.6.1 – Resultados obtidos com os programas

Exemplo	M_k (kNm)	f_{ck} (MPa)	A_s (cm ²)	A'_s (cm ²)
1	30	20	2,98	0
2	70	20	7,46	2,04
3	70	40	7,10	0
4	70	70	6,74	0

Capítulo 3

FLEXÃO NORMAL SIMPLES Dimensionamento de Seções T

3.1 – Apresentação do problema

O objetivo é dimensionar uma seção T submetida à flexão simples normal. A geometria da seção é indicada na fig. 3.1.1 A teoria é apresentada no Capítulo 5 do Volume 1.

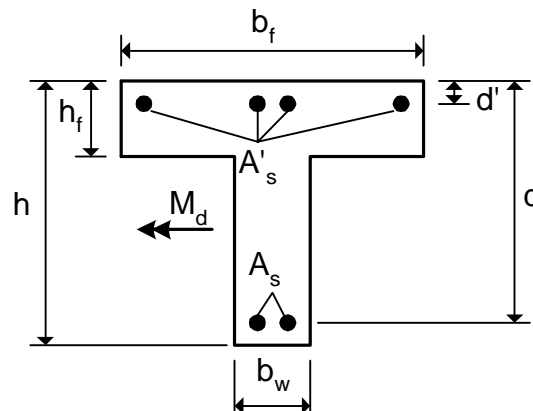


Fig. 3.1.1 - Seção T submetida à flexão simples normal

Os dados do problema são os seguintes:

- a) propriedades dos materiais: f_{ck} , f_{yk} , E_s
- b) coeficientes parciais de segurança: γ_c , γ_s , γ_f
- c) coeficiente de redistribuição de momentos: $0,75 \leq \beta \leq 1$
- d) dimensões da seção transversal: b_f , h_f , b_w , h , d , d'
- e) momento fletor de serviço: M_k (valor característico na hipótese de uma análise linear)

Os valores requeridos são as áreas de aço A_s e A'_s .

3.2 – Roteiro para programação

A) Programa principal

1) Entrada de dados

Ler as propriedades dos materiais: f_{ck} (MPa), f_{yk} (MPa), E_s (GPa)

Ler os coeficientes parciais de segurança: γ_c , γ_s , γ_f

Ler o coeficiente de redistribuição de momentos: β

Ler as dimensões da seção (em cm): b_f , h_f , b_w , h , d , d'

Ler o momento fletor de serviço em kNm: M_k

Fim da entrada de dados e início dos cálculos.

2) Parâmetros do diagrama retangular para o concreto e profundidade limite da linha neutra

- Se $f_{ck} \leq 50$ MPa:

$$\lambda = 0,8$$

$$\alpha_c = 0,85$$

$$\varepsilon_u \left(\frac{o}{oo} \right) = 3,5$$

$$\xi_{lim} = 0,8\beta - 0,35$$

- Se $f_{ck} > 50$ MPa:

$$\lambda = 0,8 - \frac{(f_{ck} - 50)}{400}$$

$$\alpha_c = 0,85 \left[1 - \frac{(f_{ck} - 50)}{200} \right]$$

$$\varepsilon_u \left(\frac{o}{oo} \right) = 2,6 + 35 \left(\frac{90 - f_{ck}}{100} \right)^4$$

$$\xi_{lim} = 0,8\beta - 0,45$$

3) Conversão de unidades para kN e cm

$$M_k = 100M_k$$

$$f_{ck} = f_{ck} / 10$$

$$f_{yk} = f_{yk} / 10$$

$$E_s = 100E_s$$

4) Resistências e momento de cálculo

$$f_{cd} = f_{ck} / \gamma_c$$

$$\sigma_{cd} = \alpha_c f_{cd}$$

$$f_{yd} = f_{yk} / \gamma_s$$

$$M_d = \gamma_f M_k$$

5) Parâmetro geométrico

$$\delta = d' / d$$

6) Parâmetros da seção T

$$\beta_f = h_f / d$$

$$\beta_w = b_w / b_f$$

7) Verificando se a seção funciona como seção T ou seção retangular

- Se $\lambda \xi_{\text{lim}} \leq \beta_f \Rightarrow$ Dimensionar como seção retangular de largura b_f e altura h . Usar o programa anterior.

8) Momento reduzido solicitante

$$\mu = \frac{M_d}{b_f d^2 \sigma_{cd}}$$

9) Momento limite

$$r_{cc \text{ lim}} = \beta_f + \beta_w (\lambda \xi_{\text{lim}} - \beta_f)$$

$$\mu_{\text{lim}} = \beta_f (1 - 0,5\beta_f) + \beta_w (\lambda \xi_{\text{lim}} - \beta_f) [1 - 0,5(\lambda \xi_{\text{lim}} + \beta_f)]$$

10) Solução com armadura simples

- Se $\mu \leq \mu_{\text{lim}}$:

Momento resistido pela mesa: $\mu_f = \beta_f (1 - 0,5\beta_f)$

$$\checkmark \text{ Se } \mu \leq \mu_f \Rightarrow \omega = 1 - \sqrt{1 - 2\mu}$$

$$\checkmark \text{ Se } \mu > \mu_f \Rightarrow \mu^* = \frac{\mu - \mu_f}{\beta_w} + \mu_f$$

$$\omega = \beta_f (1 - \beta_w) + \beta_w (1 - \sqrt{1 - 2\mu^*})$$

11) Solução com armadura dupla

- Se $\mu > \mu_{\text{lim}}$:

- ✓ Evitar o uso de armadura dupla no domínio 2:

$$\xi_a = \frac{\varepsilon_u}{\varepsilon_u + 10\%}$$

Se $\xi_{\text{lim}} < \xi_a$, colocar mensagem para o usuário aumentar as dimensões da seção transversal e parar o processamento.

- ✓ Evitar o caso em que $\xi_{\text{lim}} \leq \delta$, pois a armadura de compressão ficaria tracionada. Colocar mensagem para o usuário aumentar as dimensões da seção transversal e parar o processamento.

Tensão na armadura de compressão:

$$\varepsilon'_s = \varepsilon_u \left(\frac{\xi_{\text{lim}} - \delta}{\xi_{\text{lim}}} \right)$$

Chamar uma sub-rotina para calcular a tensão σ'_{sd} .

$$\omega' = \left(\frac{\mu - \mu_{\text{lim}}}{1 - \delta} \right) \frac{f_{yd}}{\sigma'_{sd}}$$

$$\omega = r_{cc \text{ lim}} + \frac{\mu - \mu_{\text{lim}}}{1 - \delta}$$

12) Áreas das armaduras calculadas

$$A'_s = \omega' b_f d \frac{\sigma_{cd}}{f_{yd}}$$

$$A_s = \omega b_f d \frac{\sigma_{cd}}{f_{yd}}$$

13) Armadura mínima

Voltar f_{ck} e f_{yd} para MPa:

$$f_{ck} = 10 f_{ck}$$

$$f_{yd} = 10 f_{yd}$$

- Se $f_{ck} \leq 50 \text{ MPa}$:

$$\rho_{\min} = \frac{0,078 f_{ck}^{2/3}}{f_{yd}}$$

- Se $f_{ck} > 50 \text{ MPa}$:

$$\rho_{\min} = \frac{0,5512 \ln(1 + 0,11 f_{ck})}{f_{yd}}$$

Se $\rho_{\min} < 0,15\% \rightarrow$ fazer $\rho_{\min} = 0,15\%$

Área da seção T: $A_c = b_f h_f + b_w (h - h_f)$

$$A_{s,\min} = \rho_{\min} A_c$$

Se $A_s < A_{s,\min}$, adotar $A_s = A_{s,\min}$

Imprimir ou mostrar resultado: A_s e A'_s (em cm^2)

Fim do programa principal.

B) Sub-rotina tensão

Parâmetros de entrada: E_s , ε_s , f_{yd}

Parâmetro de saída: σ_{sd}

1) Trabalhar com o valor absoluto da deformação

$$\varepsilon_{ss} = |\varepsilon_s|$$

2) Deformação de escoamento de cálculo do aço

$$\varepsilon_{yd} = f_{yd} / E_s$$

3) Cálculo da tensão

- Se $\varepsilon_{ss} < \varepsilon_{yd}$:

$$\sigma_{sd} = E_s \varepsilon_{ss}$$

- Se $\varepsilon_{ss} \geq \varepsilon_{yd}$:

$$\sigma_{sd} = f_{yd}$$

4) Acertando o sinal da tensão

Se $\varepsilon_s < 0$, fazer $\sigma_{sd} = -\sigma_{sd}$

Fim da sub-rotina.

3.3 – Programa em Fortran 90

A listagem a seguir apresenta o código do programa em Fortran 90. Os dados são lidos do arquivo DADOS.TXT. Para facilitar a comparação com os programas em Visual Basic e Visual C#, cada dado é lido em uma linha do arquivo. As mensagens de alerta e os resultados são mostrados na tela. Os dados devem ser fornecidos usando somente o ponto como separador decimal. O arquivo com a listagem do código-fonte em Fortran 90 pode ser obtido em www.editoradunas.com.br/programas.htm.

LISTAGEM DO CÓDIGO-FONTE EM FORTRAN 90

```
!
!Dimensionamento de seções T à flexão normal simples
!
!PROGRAMA PRINCIPAL
!
!Declarando variáveis reais de dupla precisão
IMPLICIT REAL*8(A-H,O-Z)
!
```

```
!Abertura do arquivo de dados
OPEN(1,FILE='DADOS.TXT')
!
!LEITURA DOS DADOS
!
!Os dados são lidos do arquivo DADOS.TXT
!Cada dado deve estar em uma linha do arquivo
!
!Resistência característica à compressão do concreto em MPa
read(1,*)fck
!
!Tensão de escoamento característica do aço em MPa
read(1,*)fyk
!Módulo de elasticidade do aço em GPa
read(1,*)es
!
!Coeficientes parciais de segurança:
!para o concreto
read(1,*)gamac
!
!para o aço
read(1,*)gamas
!
!para o momento fletor
read(1,*)gamaf
!
!Coeficiente beta de redistribuição de momentos
read(1,*)bduct
!
!Largura da mesa
read(1,*)bf
!
!Espessura da mesa
read(1,*)hf
!
!Largura da nervura
read(1,*)bw
!
!Altura da seção transversal
read(1,*)h
!
!Altura útil
read(1,*)d
!
!Parâmetro d!
read(1,*)dl
!
!Momento fletor de serviço em kNm
read(1,*)amk
!
!
!FIM DA ENTRADA DE DADOS
!
!INÍCIO DOS CÁLCULOS
!
!Parâmetros do diagrama retangular
If(fck<=50)Then
```

```

    alamb=0.8
    alfac=0.85
    eu=3.5
    qlim=0.8*bduct-0.35
Else
    alamb=0.8-(fck-50)/400
    alfac=0.85*(1-(fck-50)/200)
    eu=2.6+35*((90-fck)/100)**4
    qlim=0.8*bduct-0.45
End If
!
!Conversão de unidades: transformando para kN e cm
amk=100*amk
fck=fck/10
fyk=fyk/10
es=100*es
!
!Resistências de cálculo
fcd=fck/gamac
tcd=alfac*fcd
fyd=fyk/gamas
amd=gamaf*amk
!
!Parâmetro geométrico
delta=dl/d
!
!parâmetros da seção T
betaf=hf/d
betaw=bw/bf
!
!Verificando se a seção funciona como T
If(alamb*qlim<=betaf)Then
    !A secao funciona como seção retangular de largura bf e altura h
    !Colocar mensagem para o usuário usar o programa de seção retangular e
    !parar o processamento
    write(*,10)
10  format(/,5x,'A mesa e muito espessa. A secao funciona como secao retangular de largura bf e
altura h.')
```

goto 100

```

End If
!
!A seção será dimensionada como seção T
!Momento reduzido solicitante
ami=amd/(bf*d*d*tcd)
!
!Momento limite
!
rcclim=betaf+betaw*(alamb*qlim-betaf)
amilim=betaf*(1-0.5*betaf)
amilim=amilim+betaw*(alamb*qlim-betaf)*(1-0.5*(alamb*qlim+betaf))
!
If(ami<=amilim)Then
    !
    !Armadura simples
    !
    ! Momento resistido pela mesa
    amif=betaf*(1-0.5*betaf)
```

```

If(ami<=amif)Then
  omega=1-Sqrt(1-2*ami)
Else
  ami0=amif+(ami-amif)/betaw
  omega=betaf*(1-betaw)+betaw*(1-Sqrt(1-2*ami0))
End If
omegal=0
Else
!Armadura dupla
!
! Evitando armadura dupla no domínio 2
qsia=eu/(eu+10)
If(qlim<qsia)Then
!
! Está resultando armadura dupla no domínio 2.
! Colocar mensagem para o usuário aumentar as dimensões da seção transversal e
! parar o processamento
!
  write(*,20)
20 format(//,5x,'Resultou armadura dupla no dominio 2. Aumente as dimensoes da secao transversal')
  goto 100
End If
!
!
! Eliminando o caso em que qlim<delta
! Se isto ocorrer, a armadura de compressão estará tracionada
!
If(qlim<delta)Then
  write(*,30)
30 format(//,5x,'Aumente as dimensoes da secao transversal')
  goto 100
End If
!
! Deformação da armadura de compressão
esl=eu*(qlim-delta)/qlim
esl=esl/1000
! Tensão na armadura de compressão
! Chamar sub-rotina
Call Tensao(es,esl,fyd,tsl)
omegal=(ami-amilim)*fyd/((1-delta)*tsl)
omega=rcclim+(ami-amilim)/(1-delta)
End If
!
!Armaduras calculadas
asl=omegal*bf*d*tcd/fyd
aas=omega*bf*d*tcd/fyd
!
!Armadura mínima
a=2./3.
fck=10*fck
fyd=10*fyd
If(fck<=50)Then
  romin=0.078*(fck**a)/fyd
Else
  romin=0.5512*Log(1+0.11*fck)/fyd
End If
If (romin<0.0015)Then

```

```

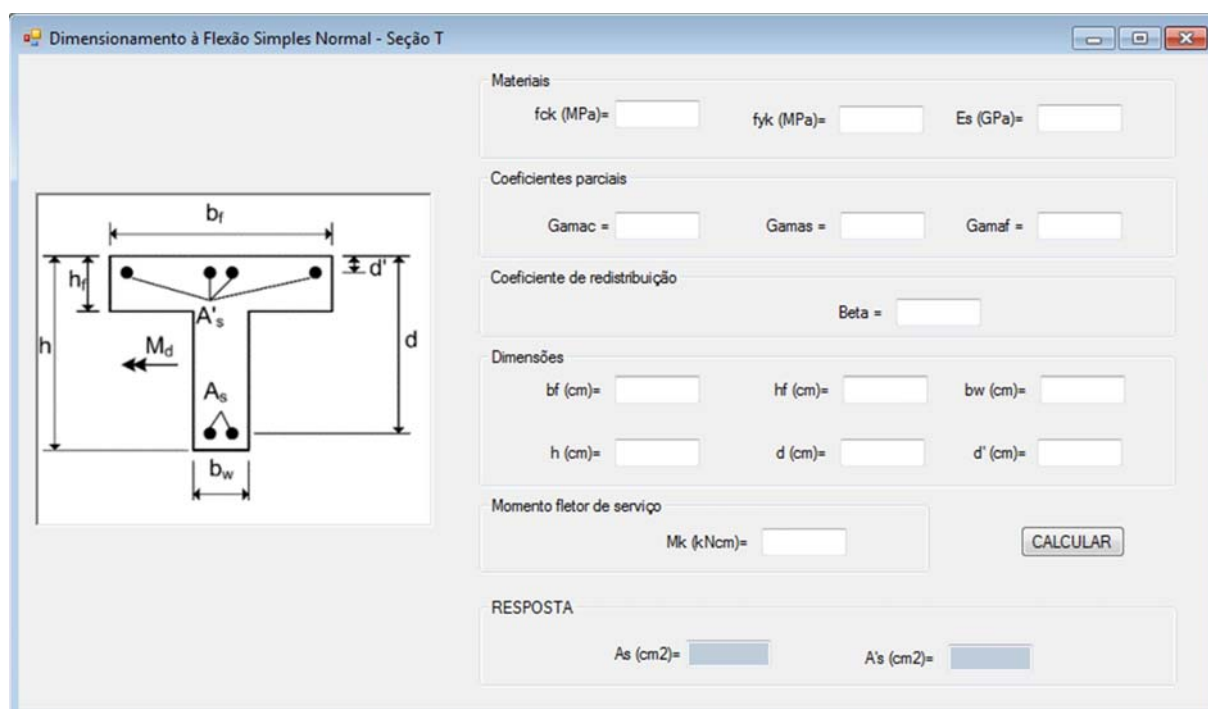
    romin=0.0015
End If
!Área da seção transversal
ac=bf*hf+bw*(h-hf)
asmin=romin*ac
If (aas<asmin)Then
    aas=asmin
End If
!
!MOSTRAR O RESULTADO
!Área da armadura tracionada: aas
!Área da armadura comprimida: asl
!
write(*,40)aas
40 format(//,5x,'Area da armadura tracionada=',f6.2)
write(*,50)asl
50 format(//,5x,'Area da armadura comprimida=',f6.2)
100 aux=0 !terminando o programa
End
!
!
SUBROUTINE Tensao(es,esl,fyd,tsl)
IMPLICIT REAL*8(A-H,O-Z)
!
!Calcula a tensão no aço
!es = módulo de elasticidade do aço em kN/cm2
!esl = deformação de entrada
!fyd = tensão de escoamento de cálculo em kN/cm2
!tsl = tensão de saída em kN/cm2
!
!Trabalhando com deformação positiva
ess=Abs(esl)
eyd=fyd/es
If(ess<eyd)Then
    tsl=es*ess
Else
    tsl=fyd
End If
!Trocando o sinal se necessário
If(esl<0)Then
    tsl=-tsl
End If
Return
End

```

3.4 – Programa em Visual Basic 2017

Na fig. 3.4.1 apresenta-se o formulário do programa escrito em Visual Basic 2017. Os dados podem ser fornecidos usando o ponto ou a vírgula como separador decimal.

Para criar o projeto, procede-se como descrito na seção 1.4 do capítulo 1. Utilizando a Caixa de Ferramentas, são introduzidos os controles no Form1. A propriedade Text do Form1 é “Dimensionamento à Flexão Simples Normal – Seção T”, como aparece no topo do formulário.



Dimensionamento à Flexão Simples Normal - Seção T

Materiais

fck (MPa)= fyk (MPa)= Es (GPa)=

Coefficientes parciais

Gamac = Gamas = Gamaf =

Coefficiente de redistribuição

Beta =

Dimensões

bf (cm)= hf (cm)= bw (cm)=

h (cm)= d (cm)= d' (cm)=

Momento fletor de serviço

Mk (kNcm)=

CALCULAR

RESPOSTA

As (cm2)= A's (cm2)=

Fig. 3.4.1 – Formulário do programa em Visual Basic 2017

O formulário contém os seguintes controles:

Tipo	Nome	Descrição
PictureBox	PictureBox1	Imagem da seção
Label	Label1	fck (MPa) =
TextBox	TextBox1	Valor de fck
Label	Label2	fyk (MPa) =
TextBox	TextBox2	Valor de fyk
Label	Label3	Es (GPa) =
TextBox	TextBox3	Valor de Es
Label	Label4	Gamac =
TextBox	TextBox4	Valor de Gamac
Label	Label5	Gamas =
TextBox	TextBox5	Valor de Gamas
Label	Label6	Gamaf =
TextBox	TextBox6	Valor de Gamaf
Label	Label7	Beta =
TextBox	TextBox7	Valor de Beta
Label	Label8	bf (cm) =
TextBox	TextBox8	Valor de bf
Label	Label9	hf (cm) =
TextBox	TextBox9	Valor de hf
Label	Label10	bw (cm) =
TextBox	TextBox10	Valor de bw
Label	Label11	h (cm) =
TextBox	TextBox11	Valor de h
Label	Label12	d (cm) =
TextBox	TextBox12	Valor de d
Label	Label13	d' =
TextBox	TextBox13	Valor de d'
Label	Label14	Mk (kNm) =
TextBox	TextBox14	Valor de Mk

Label	Label15	As (cm2) =
TextBox	TextBox15	Valor de As
Label	Label16	A's (cm2) =
TextBox	TextBox16	Valor de A's
Button	Button1	CALCULAR
GroupBox	GroupBox1	Materiais
GroupBox	GroupBox2	Coeficientes parciais
GroupBox	GroupBox3	Coeficiente de redistribuição
GroupBox	GroupBox4	Dimensões
GroupBox	GroupBox5	Momento fletor de serviço
GroupBox	GroupBox6	RESPOSTA

O arquivo com a listagem do código-fonte em Visual Basic 2017 pode ser obtido em www.editoradunas.com.br/programas.htm. Após a elaboração do formulário, basta clicar com o botão direito do mouse sobre o formulário para Exibir Código e colar o código-fonte.

LISTAGEM DO CÓDIGO-FONTE EM VISUAL BASIC 2017

Public Class Form1

Public ts1 As Double

Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click

```

Dim fck, fyk, es, gamac, gamas, gamaf, bduct As Double
Dim bf, hf, bw, h, d, dl, delta, amk As Double
Dim alamb, alfac, eu, qlim, ami, amilim As Double
Dim fcd, tcd, fyd, amd As Double
Dim betaf, betaw, rclim, amif, ami0, omega, omegal As Double
Dim aas, asl, qsia, esl, romin, asmin, a, ac As Double

```

'Esta sub-rotina corresponde ao botão CALCULAR.

'Dimensionamento de seções T à flexão normal simples

'ENTRADA DE DADOS

'Os dados são lidos das caixas de texto do formulário

'Resistência característica à compressão do concreto em MPa

'Substituindo a vírgula por ponto

TextBox1.Text = Replace(TextBox1.Text, ",", ".")

fck = Val(TextBox1.Text)

'Tensão de escoamento característica do aço em MPa

TextBox2.Text = Replace(TextBox2.Text, ",", ".")

fyk = Val(TextBox2.Text)

'Módulo de elasticidade do aço em GPa

TextBox3.Text = Replace(TextBox3.Text, ",", ".")

es = Val(TextBox3.Text)


```

'
'Coeficientes parciais de segurança:
'para o concreto
TextBox4.Text = Replace(TextBox4.Text, ",", ".")
gamac = Val(TextBox4.Text)
'
'para o aço
TextBox5.Text = Replace(TextBox5.Text, ",", ".")
gamas = Val(TextBox5.Text)
'
'para o momento fletor
TextBox6.Text = Replace(TextBox6.Text, ",", ".")
gamaf = Val(TextBox6.Text)
'
'Coeficiente beta de redistribuição de momentos
TextBox7.Text = Replace(TextBox7.Text, ",", ".")
bduct = Val(TextBox7.Text)
'
'Largura da mesa
TextBox8.Text = Replace(TextBox8.Text, ",", ".")
bf = Val(TextBox8.Text)
'
'Espessura da mesa
TextBox9.Text = Replace(TextBox9.Text, ",", ".")
hf = Val(TextBox9.Text)
'
'Largura da nervura
TextBox10.Text = Replace(TextBox10.Text, ",", ".")
bw = Val(TextBox10.Text)
'
'Altura da seção transversal
TextBox11.Text = Replace(TextBox11.Text, ",", ".")
h = Val(TextBox11.Text)
'
'Altura útil
TextBox12.Text = Replace(TextBox12.Text, ",", ".")
d = Val(TextBox12.Text)
'
'Parâmetro d'
TextBox13.Text = Replace(TextBox13.Text, ",", ".")
dl = Val(TextBox13.Text)
'
'Momento fletor de serviço em kNm
TextBox14.Text = Replace(TextBox14.Text, ",", ".")
amk = Val(TextBox14.Text)
'
'
'FIM DA ENTRADA DE DADOS
'
'INÍCIO DOS CÁLCULOS
'
'Parâmetros do diagrama retangular
If (fck <= 50) Then
    alamb = 0.8
    alfac = 0.85
    eu = 3.5
    qlim = 0.8 * bduct - 0.35

```

```

Else
    alamb = 0.8 - (fck - 50) / 400
    alfac = 0.85 * (1 - (fck - 50) / 200)
    eu = 2.6 + 35 * ((90 - fck) / 100) ^ 4
    qlim = 0.8 * bduct - 0.45
End If
'
'Conversão de unidades: transformando para kN e cm
amk = 100 * amk
fck = fck / 10
fyk = fyk / 10
es = 100 * es
'
'Resistências de cálculo
fcd = fck / gamac
tcd = alfac * fcd
fyd = fyk / gamas
amd = gamaf * amk
'
'Parâmetro geométrico
delta = dl / d
'
'parâmetros da seção T
betaf = hf / d
betaw = bw / bf
'
'Verificando se a seção funciona como T
If (alamb * qlim <= betaf) Then
    'A seção funciona como seção retangular de largura bf e altura h
    'Colocar mensagem para o usuário usar o programa de seção retangular e parar o
    processamento
    MsgBox("A mesa é muito espessa. A seção funciona como seção retangular de largura bf e
    altura h.", vbOKOnly + vbExclamation)
    Exit Sub
End If
'
'A seção será dimensionada como seção T
'Momento reduzido solicitante
ami = amd / (bf * d * d * tcd)
'
'Momento limite
'
rcclim = betaf + betaw * (alamb * qlim - betaf)
amilim = betaf * (1 - 0.5 * betaf)
amilim = amilim + betaw * (alamb * qlim - betaf) * (1 - 0.5 * (alamb * qlim + betaf))
'
If (ami <= amilim) Then
    '
    'Armadura simples
    '
    ' Momento resistido pela mesa
    amif = betaf * (1 - 0.5 * betaf)
    If (ami <= amif) Then
        omega = 1 - Math.Sqrt(1 - 2 * ami)
    Else
        ami0 = amif + (ami - amif) / betaw
    End If
End If

```

```

    omega = betaf * (1 - betaw) + betaw * (1 - Math.Sqrt(1 - 2 * ami0))
End If
omegal = 0
Else
    'Armadura dupla
    '
    ' Evitando armadura dupla no domínio 2
    qsia = eu / (eu + 10)
    If (qlim < qsia) Then
        '
        ' Está resultando armadura dupla no domínio 2.
        ' Colocar mensagem para o usuário aumentar as dimensões da seção transversal e parar o
processamento
        '
        MsgBox("Resultou armadura dupla no domínio 2. Aumente as dimensões da seção
transversal", vbOKOnly + vbExclamation)
        Exit Sub
    End If
    '
    '
    ' Eliminando o caso em que qlim<delta
    ' Se isto ocorrer, a armadura de compressão estará tracionada
    '
    If (qlim < delta) Then
        MsgBox("Aumente as dimensões da seção transversal", vbOKOnly + vbExclamation)
        Exit Sub
    End If
    ' Deformação da armadura de compressão
    esl = eu * (qlim - delta) / qlim
    esl = esl / 1000
    ' Tensão na armadura de compressão
    ' Chamar sub-rotina
    Call Tensao(es, esl, fyd)
    omegal = (ami - amilim) * fyd / ((1 - delta) * tsl)
    omega = rcclim + (ami - amilim) / (1 - delta)
End If
'
'Armaduras calculadas
asl = omegal * bf * d * tcd / fyd
aas = omega * bf * d * tcd / fyd
'
'Armadura mínima
a = 2 / 3
fck = 10 * fck
fyd = 10 * fyd
If (fck <= 50) Then
    romin = 0.078 * (fck ^ a) / fyd
Else
    romin = 0.5512 * Math.Log(1 + 0.11 * fck) / fyd
End If
If (romin < 0.0015) Then
    romin = 0.0015
End If
'Área da seção transversal
ac = bf * hf + bw * (h - hf)
asmin = romin * ac
If (aas < asmin) Then

```

```

        aas = asmin
    End If
    '
    'Convertendo a saída para duas casas decimais
    aas = FormatNumber(aas, 2)
    asl = FormatNumber(asl, 2)
    'MOSTRAR O RESULTADO
    'Área da armadura tracionada: aas
    'Área da armadura comprimida: asl
    '
    TextBox15.Text = aas
    TextBox16.Text = asl
    '
End Sub
Private Sub Tensao(es, esl, fyd)
    Dim ess, eyd As Double
    '
    'Calcula a tensão no aço
    'es = módulo de elasticidade do aço em kN/cm2
    'esl = deformação de entrada
    'fyd = tensão de escoamento de cálculo em kN/cm2
    'tsl = tensão de saída em kN/cm2
    '
    'Trabalhando com deformação positiva
    ess = Math.Abs(esl)
    eyd = fyd / es
    If (ess < eyd) Then
        tsl = es * ess
    Else
        tsl = fyd
    End If
    'Trocando o sinal se necessário
    If (esl < 0) Then
        tsl = -tsl
    End If
End Sub
End Class

```

3.5 - Programa em Visual C# 2017

Na fig. 3.5.1 apresenta-se o formulário do programa escrito em Visual C# 2017. O formulário é inteiramente análogo ao do programa em Visual Basic 2017. Os dados podem ser fornecidos usando o ponto ou a vírgula como separador decimal. Os controles são os mesmos do programa em Visual Basic.

O arquivo com a listagem do código-fonte em Visual C# 2017 pode ser obtido em www.editoradunas.com.br/programas.htm. Após a elaboração do formulário, basta clicar com o botão direito do mouse sobre o formulário para Exibir Código e colar o código-fonte.

Dimensionamento à Flexão Simples Normal - Seção T

Diagrama da Seção T: b_f , h_f , h , M_d , A'_s , A_s , b_w , d , d'

Materials
 f_{ck} (MPa) = f_{yk} (MPa) = E_s (GPa) =

Coeficientes parciais
 γ_{mac} = γ_{mas} = γ_{maf} =

Coeficiente de redistribuição
 β =

Dimensões
 b_f (cm) = h_f (cm) = b_w (cm) =
 h (cm) = d (cm) = d' (cm) =

Momento fletor de serviço
 M_k (kNm) = **CALCULAR**

Resposta
 A_s (cm²) = A'_s (cm²) =

Fig. 3.5.1 – Formulário do programa em Visual C# 2017

LISTAGEM DO CÓDIGO-FONTE EM VISUAL C# 2017

```
using System;
using System.Globalization;
using System.Windows.Forms;

namespace FSNSTE
{
    public partial class Form1 : Form
    {
        //Declarando variáveis públicas
        public double es, esl, fyd, tsl;
        public double aas, asl, omega, omegal;

        public Form1()
        {
            InitializeComponent();
        }

        private void button1_Click(object sender, EventArgs e)
        {
            //
            double fck, fyk, gamac, gamas, gamaf, bduct;
            double bf, hf, bw, h, d, dl, delta, amk;
            double alamb, alfac, eu, qlim, ami, amilim;
            double fcd, tcd, amd;
            double betaf, betaw, rcclim, amif, ami0;
            double qsia, romin, asmin, a, ac;
            //
            //
            //Esta sub-rotina corresponde ao botão CALCULAR.
        }
    }
}
```

```
//  
//  
//  
//Dimensionamento de seções T à flexão normal simples  
//  
//ENTRADA DE DADOS  
//  
//Os dados são lidos das caixas de texto do formulário  
//  
//Resistência característica à compressão do concreto em MPa  
//Substituindo a vírgula por ponto  
textBox1.Text = textBox1.Text.Replace(",", ".");  
fck = Convert.ToDouble(textBox1.Text, CultureInfo.InvariantCulture);  
//  
//Tensão de escoamento característica do aço em MPa  
textBox2.Text = textBox2.Text.Replace(",", ".");  
fyk = Convert.ToDouble(textBox2.Text, CultureInfo.InvariantCulture);  
//  
//Módulo de elasticidade do aço em GPa  
textBox3.Text = textBox3.Text.Replace(",", ".");  
es = Convert.ToDouble(textBox3.Text, CultureInfo.InvariantCulture);  
//  
//Coeficientes parciais de segurança:  
//para o concreto  
textBox4.Text = textBox4.Text.Replace(",", ".");  
gamac = Convert.ToDouble(textBox4.Text, CultureInfo.InvariantCulture);  
//  
//para o aço  
textBox5.Text = textBox5.Text.Replace(",", ".");  
gamas = Convert.ToDouble(textBox5.Text, CultureInfo.InvariantCulture);  
//  
//para o momento fletor  
textBox6.Text = textBox6.Text.Replace(",", ".");  
gamaf = Convert.ToDouble(textBox6.Text, CultureInfo.InvariantCulture);  
//  
//Coeficiente beta de redistribuição de momentos  
textBox7.Text = textBox7.Text.Replace(",", ".");  
bduct = Convert.ToDouble(textBox7.Text, CultureInfo.InvariantCulture);  
//  
//  
//Largura da mesa  
textBox8.Text = textBox8.Text.Replace(",", ".");  
bf = Convert.ToDouble(textBox8.Text, CultureInfo.InvariantCulture);  
//  
//Espessura da mesa  
textBox9.Text = textBox9.Text.Replace(",", ".");  
hf = Convert.ToDouble(textBox9.Text, CultureInfo.InvariantCulture);  
//  
//Largura da nervura  
textBox10.Text = textBox10.Text.Replace(",", ".");  
bw = Convert.ToDouble(textBox10.Text, CultureInfo.InvariantCulture);  
//  
//Altura da seção transversal  
textBox11.Text = textBox11.Text.Replace(",", ".");  
h = Convert.ToDouble(textBox11.Text, CultureInfo.InvariantCulture);  
//  
//Altura útil
```

```

textBox12.Text = textBox12.Text.Replace(",", ".");
d = Convert.ToDouble(textBox12.Text, CultureInfo.InvariantCulture);
//
//Parâmetro d//
textBox13.Text = textBox13.Text.Replace(",", ".");
dl = Convert.ToDouble(textBox13.Text, CultureInfo.InvariantCulture);
//
//Momento fletor de serviço em kNm
textBox14.Text = textBox14.Text.Replace(",", ".");
amk = Convert.ToDouble(textBox14.Text, CultureInfo.InvariantCulture);
//
//
//FIM DA ENTRADA DE DADOS
//
//INÍCIO DOS CÁLCULOS
//
//Parâmetros do diagrama retangular
if (fck <= 50)
{
    alamb = 0.8;
    alfac = 0.85;
    eu = 3.5;
    qlim = 0.8 * bduct - 0.35;
}
else
{
    alamb = 0.8 - (fck - 50) / 400;
    alfac = 0.85 * (1 - (fck - 50) / 200);
    a = (90 - fck) / 100;
    eu = 2.6 + 35 * Math.Pow(a, 4);
    qlim = 0.8 * bduct - 0.45;
}
//
//Conversão de unidades: transformando para kN e cm
amk = 100 * amk;
fck = fck / 10;
fyk = fyk / 10;
es = 100 * es;
//
//Resistências de cálculo
fcd = fck / gamac;
tcd = alfac * fcd;
fyd = fyk / gamas;
amd = gamaf * amk;
//
//Parâmetro geométrico
delta = dl / d;
//
//Parâmetros da seção T
betaf = hf / d;
betaw = bw / bf;
//
//Verificando se a seção funciona como T
if (alamb * qlim <= betaf)
{
    //A seção funciona como seção retangular de largura bf e altura h

```

```

//Colocar mensagem para o usuário usar o programa de seção retangular e parar o
processamento
//
    MessageBox.Show("A mesa é muito espessa. A seção funciona como seção retangular de
largura bf e altura h.");
    return;
}
//
//A seção será dimensionada como seção T
//Momento reduzido solicitante
ami = amd / (bf * d * d * tcd);
//
//Momento limite
//
rcclim = betaf + betaw * (alamb * qlim - betaf);
amilim = betaf * (1 - 0.5 * betaf);
amilim = amilim + betaw * (alamb * qlim - betaf) * (1 - 0.5 * (alamb * qlim + betaf));
//
if (ami <= amilim)
{
    //
    //Armadura simples
    //
    //Momento resistido pela mesa
    amif = betaf * (1 - 0.5 * betaf);
    if (ami <= amif)
    {
        omega = 1 - Math.Sqrt(1 - 2 * ami);
    }
    else
    {
        ami0 = amif + (ami - amif) / betaw;
        omega = betaf * (1 - betaw) + betaw * (1 - Math.Sqrt(1 - 2 * ami0));
    }
    omegal = 0;
}
if (ami > amilim)
{
    //
    //Evitando armadura dupla no domínio 2
    qsia = eu / (eu + 10);
    if (qlim < qsia)
    {
        //Armadura dupla
        //
        // Está resultando armadura dupla no domínio 2.
        //Colocar mensagem para o usuário aumentar as dimensões da seção transversal e parar o
processamento
        //
        MessageBox.Show("Resultou armadura dupla no domínio 2. Aumente as dimensões da
seção transversal");
        return;
    }
    //
    // Eliminando o caso em que qlim<delta
    // Se isto ocorrer, a armadura de compressão estará tracionada
    //

```



```

    if (qlim <= delta)
    {
        //
        // Colocar mensagem para o usuário aumentar as dimensões da seção transversal e parar o
processamento
        //
        MessageBox.Show("Aumente as dimensões da seção transversal");
        return;
    }
    //
    // Deformação da armadura de compressão
    esl = eu * (qlim - delta) / qlim;
    esl = esl / 1000;
    // Tensão na armadura de compressão
    // Chamar Sub-rotina
    Tensao();
    omegal = (ami - amilim) * fyd / ((1 - delta) * tsl);
    omega = rcclim + (ami - amilim) / (1 - delta);
}
//
// Armaduras calculadas
asl = omegal * bf * d * tcd / fyd;
aas = omega * bf * d * tcd / fyd;
//
// Armadura mínima
a = 2.0 / 3.0;
fck = 10 * fck;
fyd = 10 * fyd;
if (fck <= 50)
{
    romin = 0.078 * Math.Pow(fck, a) / fyd;
}
else
{
    romin = 0.5512 * Math.Log(1 + 0.11 * fck) / fyd;
}
if (romin < 0.0015)
{
    romin = 0.0015;
}
//
// Área da seção transversal
ac = bf * hf + bw * (h - hf);
asmin = romin * ac;
if (aas < asmin)
{
    aas = asmin;
}
//
// Convertendo a saída para duas casas
// decimais
decimal saida1 = Decimal.Round(Convert.ToDecimal(aas), 2);
decimal saida2 = Decimal.Round(Convert.ToDecimal(asl), 2);
//
// MOSTRAR O RESULTADO
// Área da armadura tracionada: aas
// Área da armadura comprimida: asl

```

```
//
textBox15.Text = Convert.ToString(saida1);
textBox16.Text = Convert.ToString(saida2);
//
}
private void Tensao()
{
    double ess, eyd;
    //
    //Calcula a tensão no aço
    //es = módulo de elasticidade do aço em kN/cm2
    //esl = deformação de entrada
    //fyd = tensão de escoamento de cálculo em kN/cm2
    //tsl = tensão de saída em kN/cm2
    //
    //Trabalhando com deformação positiva
    ess = Math.Abs(esl);
    eyd = fyd / es;
    if (ess < eyd)
    {
        tsl = es * ess;
    }
    else
    {
        tsl = fyd;
    }
    //Trocando o sinal se necessário
    if (esl < 0)
    {
        tsl = -tsl;
    }
}
}
```

A tensão de escoamento característica do aço é $f_{yk} = 500$ MPa e o módulo de elasticidade é $E_s = 200$ GPa. Os coeficientes parciais de segurança são $\gamma_c = 1,4$, $\gamma_s = 1,15$ e $\gamma_f = 1,4$. O coeficiente de redistribuição é $\beta = 1$, ou seja, sem redistribuição de momentos.

Na tabela 3.6.1 apresentam-se os resultados para quatro exemplos.

Tabela 3.6.1 – Resultados obtidos com os programas

Exemplo	M_k (kNm)	f_{ck} (MPa)	A_s (cm ²)	A'_s (cm ²)
1	107,14	20	10,50	0
2	178,57	20	18,49	2,09
3	178,57	40	17,20	0
4	178,57	70	16,71	0

Capítulo 4

FLEXÃO NORMAL SIMPLES Verificação da Capacidade Resistente

4.1 – Apresentação do problema

O objetivo é determinar o momento de ruína de cálculo de uma seção retangular com várias camadas de armadura, submetida à flexão simples normal. A geometria da seção é indicada na fig. 4.1.1. A teoria é apresentada no Capítulo 6 do Volume 1.

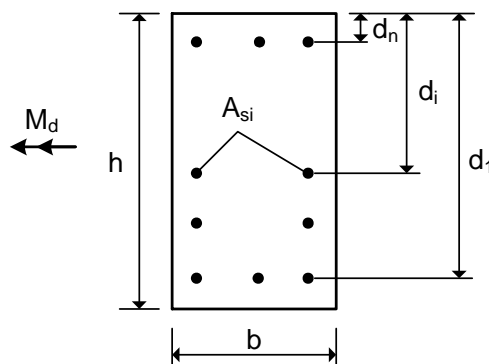


Fig. 4.1.1 - Seção retangular com várias camadas de armadura

Os dados do problema são os seguintes:

- a) propriedades dos materiais: f_{ck} , f_{yk} , E_s
- b) coeficientes parciais de segurança: γ_c , γ_s
- c) largura da seção transversal: b
- d) número de camadas de armadura: n
- e) área A_{si} e altura útil d_i de cada camada de aço

O valor requerido é o momento fletor de ruína de cálculo M_d .

4.2 – Roteiro para programação

A) Programa principal

1) Entrada de dados

Ler as propriedades dos materiais: f_{ck} (MPa), f_{yk} (MPa), E_s (GPa)

Ler os coeficientes parciais de segurança: γ_c , γ_s

Ler a largura da seção (em cm): b

Ler a área A_{si} (cm²) e a altura útil d_i (cm) de cada camada de aço.

Fim da entrada de dados e início dos cálculos.

2) Parâmetros do diagrama retangular para o concreto

- Se $f_{ck} \leq 50$ MPa:

$$\lambda = 0,8$$

$$\alpha_c = 0,85$$

$$\varepsilon_u \left(\frac{\text{o}}{\text{oo}} \right) = 3,5$$

- Se $f_{ck} > 50$ MPa:

$$\lambda = 0,8 - \frac{(f_{ck} - 50)}{400}$$

$$\alpha_c = 0,85 \left[1 - \frac{(f_{ck} - 50)}{200} \right]$$

$$\varepsilon_u \left(\frac{\text{o}}{\text{oo}} \right) = 2,6 + 35 \left(\frac{90 - f_{ck}}{100} \right)^4$$

3) Conversão de unidades para kN e cm

$$f_{ck} = f_{ck} / 10$$

$$f_{yk} = f_{yk} / 10$$

$$E_s = 100 E_s$$

4) Resistências de cálculo

$$f_{cd} = f_{ck} / \gamma_c$$

$$\sigma_{cd} = \alpha_c f_{cd}$$

$$f_{yd} = f_{yk} / \gamma_s$$

5) Processo iterativo para encontrar a linha neutra

Profundidade inicial da linha neutra: $x_i = 0$

Chamar sub-rotina para calcular a função $f_i = f(x_i)$.

Profundidade final da linha neutra: $x_f = d_1$

Chamar sub-rotina para calcular a função $f_f = f(x_f)$.

Iterar até a convergência (processo da bissecante):

Fazer: $f_k = 1$

➡ Calcular enquanto $|f_k| > 0.001$:

$$x_k = (x_i f_f - x_f f_i) / (f_f - f_i)$$

Chamar sub-rotina para calcular a função $f_k = f(x_k)$.

Calcular $prod = f_k f_i$

• Se $prod > 0$

$$x_i = x_k$$

$$f_i = f_k$$

• Se $prod < 0$

$$x_f = x_k$$

$$f_f = f_k$$

⬆ Retornar ao cálculo de x_k .

6) Cálculo do momento de ruína

Linha neutra: $x = x_k$

Braço de alavanca: $Z = d_1 - 0,5\lambda x$

Resultante de compressão: $R_{cc} = \lambda b x \sigma_{cd}$

Momento de ruína: equação (6.3.9) do Volume 1:

$$M_d = R_{cc} Z + \sum_{i=1}^n A_{si} \sigma_{si} (d_1 - d_i)$$

Fim do programa principal.

B) Sub-rotina função

Essa sub-rotina calcula o valor da função $f(x)$ dada na equação (6.3.10) do Volume 1. A profundidade da linha neutra x é um parâmetro de entrada na sub-rotina.

1) Constantes para cálculo das deformações na armadura

$$x_1 = \left(\frac{\varepsilon_u}{\varepsilon_u + 10} \right) d_1$$

• Se $x \leq x_1$, a linha neutra está no domínio 2:

$$c = 0,01 / (d_1 - x)$$

- Se $x > x_1$, a linha neutra está nos domínios 3 ou 4:

$$c = (\varepsilon_u / 1000) / x$$

2) Resultante de compressão no concreto

$$R_{cc} = \lambda b x \sigma_{cd}$$

$$f(x) = R_{cc}$$

3) Superpondo a contribuição das armaduras

➡ Somar de 1 a n: $\sum_{i=1}^n$

$$\varepsilon_{si} = c(x - d_i)$$

Chamar sub-rotina para calcular a tensão σ_{si} .

⬆ $f = f + A_{si} \sigma_{si}$

Normalizar a função $f(x)$: $f = f / (b d_1 \sigma_{cd})$

Fim da sub-rotina.

C) Sub-rotina tensão

Parâmetros de entrada: E_s , ε_s , f_{yd}

Parâmetro de saída: σ_{sd}

1) Trabalhar com o valor absoluto da deformação

$$\varepsilon_{ss} = |\varepsilon_s|$$

2) Deformação de escoamento de cálculo do aço

$$\varepsilon_{yd} = f_{yd} / E_s$$

3) Cálculo da tensão

- Se $\varepsilon_{ss} < \varepsilon_{yd}$:

$$\sigma_{sd} = E_s \varepsilon_{ss}$$

- Se $\varepsilon_{ss} \geq \varepsilon_{yd}$:

$$\sigma_{sd} = f_{yd}$$

4) Acertando o sinal da tensão

Se $\varepsilon_s < 0$, fazer $\sigma_{sd} = -\sigma_{sd}$

Fim da sub-rotina.

4.3 – Programa em Fortran 90

A listagem a seguir apresenta o código do programa em Fortran 90. Os dados são lidos do arquivo DADOS.TXT. Para facilitar a comparação com os programas em Visual Basic e Visual C#, cada dado é lido em uma linha do arquivo. As mensagens de alerta e os resultados são mostrados na tela. Os dados devem ser fornecidos usando somente o ponto como separador decimal. O arquivo com a listagem do código-fonte em Fortran 90 pode ser obtido em www.editoradunas.com.br/programas.htm.

LISTAGEM DO CÓDIGO-FONTE EM FORTRAN 90

```
!Flexão Simples Normal
!Verificação de seções retangulares com várias camadas de armadura
!O programa está limitado a 20 camadas
!
!
!PROGRAMA PRINCIPAL
!
!Declarando variáveis reais de dupla precisão
IMPLICIT REAL*8(A-H,O-Z)
COMMON /ARMADURA/ asi(20),di(20),tens(20)
!
!Abertura do arquivo de dados
OPEN(1,FILE='DADOS.TXT')
!
!LEITURA DOS DADOS
!
!Os dados são lidos do arquivo DADOS.TXT
!Cada dado deve estar em uma linha do arquivo
!
!Resistência característica à compressão do concreto em MPa
read(1,*)fck
!
!Tensão de escoamento característica do aço em MPa
read(1,*)fyk
!Módulo de elasticidade do aço em GPa
read(1,*)es
!
!Coeficientes parciais de segurança:
!para o concreto
read(1,*)gamac
!
!para o aço
read(1,*)gamas
!
!Largura da seção transversal
read(1,*)b
!
!Número de camadas de armadura
read(1,*)n
!
!Dados das camadas de armadura
!As camadas são numeradas de baixo para cima
```



```

do i=1,n
! Área de aço e altura útil
  read(1,*)asi(i),di(i)
end do
!
!
!FIM DA ENTRADA DE DADOS
!
!INÍCIO DOS CÁLCULOS
!
!Parâmetros do diagrama retangular
If(fck<=50)Then
  alamb=0.8
  alfac=0.85
  eu=3.5
Else
  alamb=0.8-(fck-50)/400
  alfac=0.85*(1-(fck-50)/200)
  eu=2.6+35*((90-fck)/100)**4
End If
!
!Conversão de unidades: transformando para kN e cm
fck=fck/10
fyk=fyk/10
es=100*es
!
!Resistências de cálculo
fcd=fck/gamac
tcd=alfac*fcd
fyd=fyk/gamas
!
!Cálculo do momento de ruína através do processo iterativo da bissecante
!
!Valor inicial para a linha neutra
xi=0
!Chamar sub-rotina
Call funcao(b,xi,n,fcd,fyd,es,rc,alfac,alamb,eu,fi)
!
!Valor final para a linha neutra
xf=di(1)
!Chamar sub-rotina
Call funcao(b,xf,n,fcd,fyd,es,rc,alfac,alamb,eu,ff)
!
!Processo iterativo da bissecante
fk=1
Do While (Abs(fk)>0.001)
  xk=(xi*ff-xf*fi)/(ff-fi)
  Call funcao(b,xk,n,fcd,fyd,es,rc,alfac,alamb,eu,fk)
  prod=fk*fi
  If(prod>0)Then
    xi=xk
    fi=fk
  Else
    xf=xk
    ff=fk
  End If
End do

```

```

!Convergência alcançada
!xk é a raiz da função f(x) dada na equação (6.3.10) do Volume 1 de Curso de
!Concreto Armado
!Momento de ruína de cálculo
x=xk
zc=di(1)-0.5*alamb*x
amu=rc*zc
Do i=1,n
    amu=amu+asi(i)*tens(i)*(di(1)-di(i))
End do

!Passando o momento para kNm
amu=amu/100
!
!
!MOSTRAR O RESULTADO
!
write(*,10)amu
10 format(//,5x,'Momento de ruína de calculo=',f6.2)
!Fim do programa
End
!
!
!
Subroutine Funcao(b,x,n,fcd,fyd,es,rc,alfac,alamb,eu,f)
IMPLICIT REAL*8(A-H,O-Z)
COMMON /ARMADURA/ asi(20),di(20),tens(20)
!
!Calcula o valor da função f(x) dada na equação (6.3.10) do Volume 1 de Curso de
!Concreto Armado
!O valor de saída é a variável f
!
tcd=alfac*fcd
!Constantes para o cálculo das deformações das camadas de armadura
xl=eu*di(1)/(eu+10)
If(x<=xl)Then
    !A linha neutra está no domínio 2
    c=0.01/(di(1)- x)
Else
    !A linha neutra está nos domínios 3 ou 4
    c=eu/(1000*x)
End If
!Resultante de compressão no concreto
rc=alamb*b*x*tcd
f=rc
!Superpondo a contribuição das armaduras
Do i=1,n
    esi=c*(x-di(i))
    Call Tensao(es,esi,fyd,tsi)
    tens(i)=tsi
    f=f+asi(i)*tsi
End do
!Transformando f em adimensional para testar a convergência
f=f/(b*di(1)*tcd)
!
Return
End

```

```

!
!
!
SUBROUTINE Tensao(es,esl,fyd,tsl)
IMPLICIT REAL*8(A-H,O-Z)
!
!Calcula a tensao no aço
!es = módulo de elasticidade do aço em kN/cm2
!esl = deformação de entrada
!fyd = tensão de escoamento de cálculo em kN/cm2
!tsl = tensão de saída em kN/cm2
!
!Trabalhando com deformação positiva
ess=Abs(esl)
eyd=fyd/es
If(ess<eyd)Then
    tsl=es*ess
Else
    tsl=fyd
End If
!Trocando o sinal se necessário
If(esl<0)Then
    tsl=-tsl
End If
Return
End

```

4.4 – Programa em Visual Basic 2017

Na fig. 4.4.1 apresenta-se o formulário do programa escrito em Visual Basic 2017. Os dados podem ser fornecidos usando o ponto ou a vírgula como separador decimal. Os dados das camadas de armadura são fornecidos em uma grade, a qual é exibida assim que o usuário digitar o número de camadas.

The screenshot shows a Windows application window titled "Flexão Simples Normal - Verificação de Seções Retangulares". On the left, there is a diagram of a rectangular cross-section with width b , total height h , effective depth d_1 , and distance from the top reinforcement to the top fiber d_n . The area of reinforcement is labeled A_{si} . The main area of the form contains several input fields grouped into sections: "Materiais" with f_{ck} (MPa), f_{yk} (MPa), and E_s (GPa); "Coeficientes parciais" with γ_{mac} and γ_{mas} ; "Largura da seção" with b (cm); "Número de camadas" with n ; and a table for reinforcement layers with columns "Camada", "Área (cm²)", and "Altura útil d (cm)". A "CALCULAR" button is located to the right of the table. At the bottom, there is a field for "Momento de ruína de cálculo" M_d (kNm).

Fig. 4.4.1 – Formulário do programa em Visual Basic 2017

Para criar o projeto, procede-se como descrito na seção 1.4 do capítulo 1. Utilizando a Caixa de Ferramentas, são introduzidos os controles no Form1. A propriedade Text do Form1 é “Flexão Simples Normal – Verificação de Seções Retangulares”, como aparece no topo do formulário.

O arquivo com a listagem do código-fonte em Visual Basic 2017 pode ser obtido em www.editoradunas.com.br/programas.htm. Após a elaboração do formulário, basta clicar com o botão direito do mouse sobre o formulário para Exibir Código e colar o código-fonte.

O formulário contém os seguintes controles:

Tipo	Nome	Descrição
PictureBox	PictureBox1	Imagem da seção
Label	Label1	fck (MPa) =
TextBox	TextBox1	Valor de fck
Label	Label2	fyk (MPa) =
TextBox	TextBox2	Valor de fyk
Label	Label3	Es (GPa) =
TextBox	TextBox3	Valor de Es
Label	Label4	Gamac =
TextBox	TextBox4	Valor de Gamac
Label	Label5	Gamas =
TextBox	TextBox5	Valor de Gamas
Label	Label6	b (cm) =
TextBox	TextBox6	Valor de b
Label	Label7	n =
TextBox	TextBox7	Valor de n
Label	Label8	Md (kNm) =
TextBox	TextBox8	Valor de Md
DataGridView	DataGridView1	Dados das camadas de aço
Button	Button1	CALCULAR
GroupBox	GroupBox1	Materiais
GroupBox	GroupBox2	Coeficientes parciais
GroupBox	GroupBox3	Largura da seção
GroupBox	GroupBox4	Número de camadas
GroupBox	GroupBox5	Momento de ruína de cálculo

Para o controle DataGridView1, Habilitar Edição. Para a coluna 1, definir a propriedade ReadOnly como True.

LISTAGEM DO CÓDIGO-FONTE EM VISUAL BASIC 2017

```
Public Class Form1
```

```
    Public f, tsl, tens(20) As Double
```

```
    Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
```

```
        Dim fck, fyk, es, gamac, gamas, b As Double
```

```
        Dim asi(20), di(20) As Double
```

```
        Dim alamb, alfac, eu, fcd, tcd, fyd As Double
```

```
        Dim xi, xf, xk, x, prod, rc, zc, amu As Double
```

```
        Dim fi, ff, fk As Double
```

```
        Dim n As Integer
```

```
        Dim valor As String
```

```

'
'
'Esta sub-rotina corresponde ao botão CALCULAR.
'
'
'Flexão Simples Normal
'Verificação de seções retangulares com várias camadas de armadura
'O programa está limitado a 20 camadas
'
'
'ENTRADA DE DADOS
'
'Os dados são lidos das caixas de texto do formulário e da grade.
'Resistência característica à compressão do concreto em MPa
'Substituindo a vírgula por ponto
TextBox1.Text = Replace(TextBox1.Text, ",", ".")
fck = Val(TextBox1.Text)
'
'Tensão de escoamento característica do aço em MPa
TextBox2.Text = Replace(TextBox2.Text, ",", ".")
fyk = Val(TextBox2.Text)
'
'Módulo de elasticidade do aço em GPa
TextBox3.Text = Replace(TextBox3.Text, ",", ".")
es = Val(TextBox3.Text)
'
'Coeficientes parciais de segurança:
'para o concreto
TextBox4.Text = Replace(TextBox4.Text, ",", ".")
gamac = Val(TextBox4.Text)
'para o aço
TextBox5.Text = Replace(TextBox5.Text, ",", ".")
gamas = Val(TextBox5.Text)
'
'Largura da seção transversal
TextBox6.Text = Replace(TextBox6.Text, ",", ".")
b = Val(TextBox6.Text)
'
'Número de camadas de armadura
n = Val(TextBox7.Text)
'
'Dados das camadas de armadura
'As camadas são numeradas de baixo para cima
For i = 0 To n - 1
'Convertendo ponto em vírgula no DataGridView
'O usuário pode usar ponto ou vírgula como separador decimal
'Segunda coluna (áreas de aço)
valor = DataGridView1.Item(1, i).Value.ToString
valor = valor.Replace(".", ",")
DataGridView1.Item(1, i).Value = valor
'Segunda coluna (alturas úteis)
valor = DataGridView1.Item(2, i).Value.ToString
valor = valor.Replace(".", ",")
DataGridView1.Item(2, i).Value = valor
'Lendo os dados do DataGridView
asi(i) = DataGridView1.Item(1, i).Value
di(i) = DataGridView1.Item(2, i).Value

```

```

Next
,
,
'FIM DA ENTRADA DE DADOS
,
'INÍCIO DOS CÁLCULOS
,
,
'Parâmetros do diagrama retangular
If (fck <= 50) Then
    alamb = 0.8
    alfac = 0.85
    eu = 3.5
Else
    alamb = 0.8 - (fck - 50) / 400
    alfac = 0.85 * (1 - (fck - 50) / 200)
    eu = 2.6 + 35 * ((90 - fck) / 100) ^ 4
End If
,
,
'Conversão de unidades: transformando para kN e cm
fck = fck / 10
fyk = fyk / 10
es = 100 * es
,
,
'Resistências de cálculo
fcd = fck / gamac
tcd = alfac * fcd
fyd = fyk / gamas
,
,
'Cálculo do momento de ruína através do processo iterativo da bissecante
,
'Valor inicial para a linha neutra
xi = 0
'Chamar sub-rotina
Call Funcao(b, xi, n, tcd, fyd, es, alfac, alamb, eu, asi, di)
fi = f
,
'Valor final para a linha neutra
xf = di(0)
'Chamar sub-rotina
Call Funcao(b, xf, n, tcd, fyd, es, alfac, alamb, eu, asi, di)
ff = f
,
'Processo iterativo da bissecante
fk = 1
Do While (Math.Abs(fk) > 0.001)
    xk = (xi * ff - xf * fi) / (ff - fi)
    Call Funcao(b, xk, n, tcd, fyd, es, alfac, alamb, eu, asi, di)
    fk = f
    prod = fk * fi
    If (prod > 0) Then
        xi = xk
        fi = fk
    Else
        xf = xk
        ff = fk
    End If
End Do

```

```

    End If
Loop
'Convergência alcançada
'xk é a raiz da função f(x) dada na equação (6.3.10) do Volume 1 de Curso de Concreto Armado
'Momento de ruína de cálculo
x = xk
rc = alamb * b * x * tcd
zc = di(0) - 0.5 * alamb * x
amu = rc * zc
For i = 0 To n - 1
    amu = amu + asi(i) * tens(i) * (di(0) - di(i))
Next
'Passando o momento para kNm
amu = amu / 100
'
'Convertendo a saída para duas casas decimais
amu = FormatNumber(amu, 2)
'
TextBox8.Text = amu
'
End Sub
Private Sub Funcao(b, x, n, tcd, fyd, es, alfac, alamb, eu, asi, di)
    Dim xl, c, rc, esi As Double
    '
    'Calcula o valor da função f(x) dada na equação (6.3.10) do Volume 1 de Curso de Concreto
Armado
    'O valor de saída é a variável f
    '
    'Constantes para o cálculo das deformações das camadas de armadura
xl = eu * di(0) / (eu + 10)
If (x <= xl) Then
    'A linha neutra está no domínio 2
    c = 0.01 / (di(0) - x)
Else
    'A linha neutra está nos domínios 3 ou 4
    c = eu / (1000 * x)
End If
'Resultante de compressão no concreto
rc = alamb * b * x * tcd
f = rc
'Superpondo a contribuição das armaduras
For i = 0 To n - 1
    esi = c * (x - di(i))
    Call Tensao(es, esi, fyd)
    tens(i) = tsl
    f = f + asi(i) * tsl
Next
'Transformando f em adimensional para testar a convergência
f = f / (b * di(0) * tcd)
'
End Sub
Private Sub Tensao(es, esl, fyd)
    Dim ess, eyd As Double
    '
    'Calcula a tensão no aço
'es = módulo de elasticidade do aço em kN/cm2

```

```

'esl = deformação de entrada
'fyd = tensão de escoamento de cálculo em kN/cm2
'tsl = tensão de saída em kN/cm2
'

'Trabalhando com deformação positiva
ess = Math.Abs(esl)
eyd = fyd / es
If (ess < eyd) Then
    tsl = es * ess
Else
    tsl = fyd
End If
'Trocando o sinal se necessário
If (esl < 0) Then
    tsl = -tsl
End If
End Sub

Private Sub TextBox7_TextChanged(sender As Object, e As EventArgs) Handles
    TextBox7.TextChanged
    Dim i, n As Integer
    'Essa sub-rotina corresponde à caixa de texto onde é fornecido o número de camadas de armadura
    'Ela faz o dimensionamento dinâmico da DataGridView1
    '
    n = Val(TextBox7.Text)
    If (n > 0) Then
        'Limpando o DataGridView1
        DataGridView1.Rows.Clear()
        'Criando o DataGridView1 com n linhas
        For i = 0 To n - 1
            DataGridView1.Rows.Add()
        Next
        'Centralizando o texto nas 3 colunas
        For j = 0 To 2
            DataGridView1.Columns(j).DefaultCellStyle.Alignment =
DataGridViewContentAlignment.MiddleCenter
        Next
        'Preenchendo a primeira coluna
        For i = 0 To n - 1
            DataGridView1.Item(0, i).Value = i + 1
        Next
    End If
End Sub

End Class

```

4.5 - Programa em Visual C# 2017

Na fig. 4.5.1 apresenta-se o formulário do programa escrito em Visual C# 2017. O formulário é inteiramente análogo ao do programa em Visual Basic 2017. Os dados podem ser fornecidos usando o ponto ou a vírgula como separador decimal. Os controles são os mesmos do programa em Visual Basic.

O arquivo com a listagem do código-fonte em Visual C# 2017 pode ser obtido em www.editoradunas.com.br/programas.htm. Após a elaboração do formulário, basta clicar com o botão direito do mouse sobre o formulário para Exibir Código e colar o código-fonte.

Fig. 4.5.1 – Formulário do programa em Visual C# 2017

LISTAGEM DO CÓDIGO-FONTE EM VISUAL C# 2017

```
using System;
using System.Globalization;
using System.Windows.Forms;

namespace VSRET
{
    public partial class Form1 : Form
    {
        //Declarando variáveis públicas
        public double[] tens = new double[20];
        public double[] asi = new double[20];
        public double[] di = new double[20];
        public double es, esl, fyd, rc, f, tsl;
        public double b, x, n, fcd, tcd, alfac, alamb, eu;
        public double fck, fyk, gamac, gamas;
        public double xi, xf, xk, prod, zc, amu;
        public double a, fi, ff, fk;
        public string valor;
        //
        public Form1()
        {
            InitializeComponent();
        }

        private void button1_Click(object sender, EventArgs e)
        {
            int i, n;
            //
            //
        }
    }
}
```

```

//Esta sub-rotina corresponde ao botão CALCULAR.
//
//
//Flexão Simples Normal
//Verificação de seções retangulares com várias camadas de armadura
//O programa está limitado a 20 camadas
//
//
//ENTRADA DE DADOS
//
//Os dados são lidos das caixas de texto do formulário e da grade.
//Resistência característica à compressão do concreto em MPa
//Substituindo a vírgula por ponto
textBox1.Text = textBox1.Text.Replace(",", ".");
fck = Convert.ToDouble(textBox1.Text, CultureInfo.InvariantCulture);
//
//Tensão de escoamento característica do aço em MPa
textBox2.Text = textBox2.Text.Replace(",", ".");
fyk = Convert.ToDouble(textBox2.Text, CultureInfo.InvariantCulture);
//
//Módulo de elasticidade do aço em GPa
textBox3.Text = textBox3.Text.Replace(",", ".");
es = Convert.ToDouble(textBox3.Text, CultureInfo.InvariantCulture);
//
//Coeficientes parciais de segurança:
//para o concreto
textBox4.Text = textBox4.Text.Replace(",", ".");
gamac = Convert.ToDouble(textBox4.Text, CultureInfo.InvariantCulture);
//
//para o aço
textBox5.Text = textBox5.Text.Replace(",", ".");
gamas = Convert.ToDouble(textBox5.Text, CultureInfo.InvariantCulture);
//
//Largura da seção transversal
textBox6.Text = textBox6.Text.Replace(",", ".");
b = Convert.ToDouble(textBox6.Text, CultureInfo.InvariantCulture);
//
//Número de camadas de armadura
n = Convert.ToInt16(textBox7.Text, CultureInfo.InvariantCulture);
//Dados das camadas de armadura
//As camadas são numeradas de baixo para cima
for (i = 0; i <= n-1; i++)
{
    //Convertendo ponto em vírgula no DataGridView
    //O usuário pode usar ponto ou vírgula como separador decimal
    //Segunda coluna (áreas de aço)
    valor = Convert.ToString(dataGridView1[1, i].Value, CultureInfo.InvariantCulture);
    valor = valor.Replace(",", ".");
    dataGridView1[1, i].Value = valor;
    //Segunda coluna (alturas úteis)
    valor = Convert.ToString(dataGridView1[2, i].Value, CultureInfo.InvariantCulture);
    valor = valor.Replace(",", ".");
    dataGridView1[2, i].Value = valor;
    //Lendo os dados do DataGridView
    asi[i] = Convert.ToDouble(dataGridView1[1, i].Value, CultureInfo.InvariantCulture);
    di[i] = Convert.ToDouble(dataGridView1[2, i].Value, CultureInfo.InvariantCulture);
}

```

```

//
//
//FIM DA ENTRADA DE DADOS
//
//INÍCIO DOS CÁLCULOS
//
//
//Parâmetros do diagrama retangular
if (fck <= 50)
{
    alamb = 0.8;
    alfac = 0.85;
    eu = 3.5;
}
else
{
    alamb = 0.8 - (fck - 50) / 400;
    alfac = 0.85 * (1 - (fck - 50) / 200);
    a = (90 - fck) / 100;
    eu = 2.6 + 35 * Math.Pow(a, 4);
}
//
//Conversão de unidades: transformando para kN e cm
fck = fck / 10;
fyk = fyk / 10;
es = 100 * es;
//
//Resistências de cálculo
fcd = fck / gamac;
tcd = alfac * fcd;
fyd = fyk / gamas;
//
//
//Cálculo do momento de ruína através do processo iterativo da bissecante
//
//Valor inicial para a linha neutra
xi = 0;
//Chamar sub-rotina
x = xi;
Funcao();
fi = f;
//
//Valor final para a linha neutra
xf = di[0];
//Chamar sub-rotina
x = xf;
Funcao();
ff = f;
//
//Processo iterativo da bissecante
fk = 1;
while(Math.Abs(fk) > 0.001)
{
    xk = (xi * ff - xf * fi) / (ff - fi);
    x = xk;
    Funcao();
    fk = f;
}

```

```

    prod = fk * fi;
    if(prod > 0)
    {
        xi = xk;
        fi = fk;
    }
    else
    {
        xf = xk;
        ff = fk;
    }
}
//Convergência alcançada
//x=xk é a raiz da função f(x) dada na equação (6.3.10) do Volume 1 de Curso de Concreto
Armado
//Momento de ruína de cálculo
rc = alamb * b * x * tcd;
zc = di[0] - 0.5 * alamb * x;
amu = rc * zc;
for (i = 0; i <= n-1; i++)
{
    amu = amu + asi[i] * tens[i] * (di[0] - di[i]);
}
//Passando o momento para kNm
amu = amu / 100;
//
//Convertendo a saída para duas casas decimais
decimal saida = Decimal.Round(Convert.ToDecimal(amu), 2);
//
textBox8.Text = Convert.ToString(saida);
//
}
//
//
private void Funcao()
{
    double x1, c;
    int i,n;
    n = Convert.ToInt16(textBox7.Text, CultureInfo.InvariantCulture);
    //
    //Calcula o valor da função f(x) dada na equação (6.3.10) do Volume 1 de Curso de Concreto
Armado
    //O valor de saída é a variável f
    //
    //Constantes para o cálculo das deformações das camadas de armadura
    x1 = eu * di[0] / (eu + 10);
    if(x<=x1)
    {
        //A linha neutra está no domínio 2
        c = 0.01 / (di[0] - x);
    }
    else
    {
        //A linha neutra está nos domínios 3 ou 4
        c = eu / (1000 * x);
    }
    //Resultante de compressão no concreto

```

```

rc = alamb * b * x * tcd;
f = rc;
//Superpondo a contribuição das armaduras
for (i = 0; i <= n-1; i++)
{
    esl = c * (x - di[i]);
    Tensao();
    tens[i] = tsl;
    f = f + asi[i] * tsl;
}
//Transformando f em adimensional para testar a convergência
f = f / (b * di[0] * tcd);
//
}
//
//
private void Tensao()
{
    double ess, eyd;
    //
    //Calcula a tensão no aço
    //es = módulo de elasticidade do aço em kN/cm2
    //esl = deformação de entrada
    //fyd = tensão de escoamento de cálculo em kN/cm2
    //tsl = tensão de saída em kN/cm2
    //
    //Trabalhando com deformação positiva
    ess = Math.Abs(esl);
    eyd = fyd / es;
    if (ess < eyd)
    {
        tsl = es * ess;
    }
    else
    {
        tsl = fyd;
    }
    //Trocando o sinal se necessário
    if (esl < 0)
    {
        tsl = -tsl;
    }
}
//
//
private void textBox7_TextChanged(object sender, EventArgs e)
{
    int i, j, n;
    //Essa sub-rotina corresponde à caixa de texto onde é fornecido o número de camadas de
armadura
    //Ela faz o dimensionamento dinâmico do DataGridView1
    //
    n = Convert.ToInt16(textBox7.Text, CultureInfo.InvariantCulture);
    if (n > 0)
    {
        //Limpando o DataGridView1
        dataGridView1.Rows.Clear();
    }
}

```

```

//Criando o DataGridView1 com n linhas
for (i=0; i<=n-1; i++)
{
    dataGridView1.Rows.Add();
}
//Centralizando o texto nas 3 colunas
for (j = 0; j <= 2; j++)
{
    dataGridView1.Columns[j].DefaultCellStyle.Alignment =
DataGridViewContentAlignment.MiddleCenter;
}
//Preenchendo a primeira coluna
for(i=0; i<=n-1; i++)
{
    dataGridView1[0, i].Value = i + 1;
}
}
//
//
}
}

```

4.6 – Exemplos para testar os programas

Os resultados apresentados a seguir servem para o leitor testar os programas. Os exemplos correspondem à seção retangular da fig. 4.6.1. Esses exemplos foram resolvidos manualmente no Capítulo 6 do Volume 1.

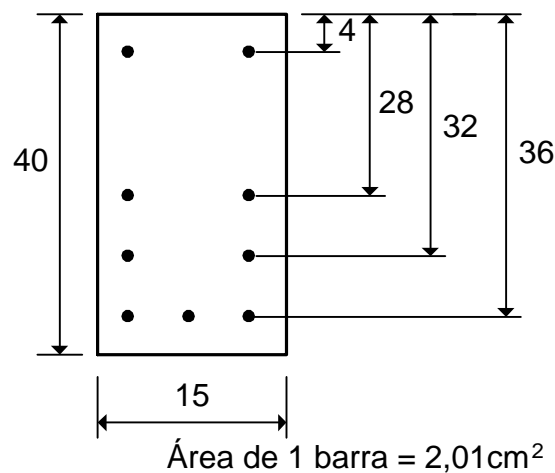


Fig. 4.6.1 - Dimensões da seção retangular

A tensão de escoamento característica do aço é $f_{yk} = 500 \text{ MPa}$ e o módulo de elasticidade é $E_s = 200 \text{ GPa}$. Os coeficientes parciais de segurança são $\gamma_c = 1,4$ e $\gamma_s = 1,15$.

O exemplo 1 corresponde à seção com quatro camadas de armadura, conforme a fig. 4.6.1. O exemplo 2 corresponde à seção com duas camadas de armadura, sendo toda a armadura tracionada concentrada na camada inferior. No exemplo 3, a armadura tracionada é concentrada no seu centroide.

Na tabela 4.6.1 apresentam-se os resultados dos três exemplos.

Tabela 4.6.1 – Resultados obtidos com os programas

Exemplo	n	A_{si} (cm ²)	d_i (cm)	M_d (kNm)
1	4	6,03	36	129,14
		4,02	32	
		4,02	28	
		4,02	4	
2	2	14,07	36	147,23
		4,02	4	
3	2	14,07	32,57	125,54
		4,02	4	

Capítulo 5

ESFORÇO CORTANTE

5.1 – Apresentação do problema

O objetivo é dimensionar os estribos verticais para uma seção retangular ou uma seção T, como indicadas na fig. 5.1.1 A teoria é apresentada no Capítulo 7 do Volume 1.

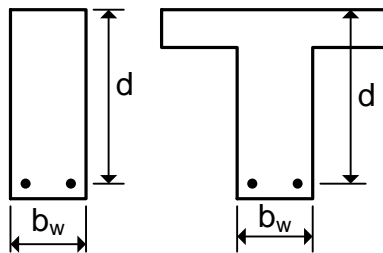


Fig. 5.1.1 - Seção retangular e seção T

Os dados do problema são os seguintes:

- a) propriedades dos materiais: f_{ck} , f_{yk}
- b) coeficientes parciais de segurança: γ_c , γ_s , γ_f
- c) largura da nervura seção transversal: b_w
- d) altura útil: d
- e) esforço cortante de serviço: V_k

O valor requerido é a área de estribos A_{sw} em cm^2/m .

5.2 – Roteiro para programação

1) Entrada de dados

Ler as propriedades dos materiais: f_{ck} (MPa), f_{yk} (MPa)

Ler os coeficientes parciais de segurança: γ_c , γ_s , γ_f

Ler a largura da seção (em cm): b_w

Ler a altura útil da seção (em cm): d

Ler o esforço cortante de serviço (em kN): V_k

Fim da entrada de dados e início dos cálculos.

2) Resistências e esforço cortante de cálculo

$$f_{cd} = f_{ck} / \gamma_c$$

$$f_{yd} = f_{yk} / \gamma_s$$

$$V_d = \gamma_f V_k$$

3) Tensão convencional de cisalhamento

$$\tau_{wd} = \frac{V_d}{b_w d}$$

Passar para MPa: $\tau_{wd} = 10 \tau_{wd}$

4) Tensão de cisalhamento última

$$\alpha_v = 1 - \frac{f_{ck}}{250}$$

$$\tau_{wu} = 0,27 \alpha_v f_{cd}$$

5) Verificação do esmagamento das bielas

- Se $\tau_{wd} > \tau_{wu}$:

Está ocorrendo esmagamento das bielas. Colocar mensagem para o usuário aumentar as dimensões da seção transversal e parar o processamento.

6) Cálculo da armadura

- Se $f_{ck} \leq 50$ MPa:

$$\tau_c = 0,126 (f_{ck})^{2/3} / \gamma_c$$

- Se $f_{ck} > 50$ MPa:

$$\tau_c = 0,8904 \ln(1 + 0,11 f_{ck}) / \gamma_c$$

Obs: Se for adotado $\gamma_c = 1,4$, essas relações ficam iguais às equações (7.4.6) e (7.4.7) do Volume 1.

$$\tau_d = 1,11 (\tau_{wd} - \tau_c)$$

- Se $\tau_d < 0$: fazer $\tau_d = 0$
- Se $f_{yd} > 435$: fazer $f_{yd} = 435$

$$A_{sw} = 100 b_w \frac{\tau_d}{f_{yd}}$$

7) Armadura mínima

- Se $f_{yk} > 500$: fazer $f_{yk} = 500$

- Se $f_{ck} \leq 50$ MPa:

$$f_{ctm} = 0,30(f_{ck})^{2/3}$$

- Se $f_{ck} > 50$ MPa:

$$f_{ctm} = 2,12 \ln(1 + 0,11 f_{ck})$$

$$\rho_{w,\min} = 0,2 \frac{f_{ctm}}{f_{yk}}$$

$$A_{sw,\min} = \rho_{w,\min} 100 b_w$$

Se $A_{sw} < A_{sw,\min}$: fazer $A_{sw} = A_{sw,\min}$.

Imprimir ou mostrar o resultado: A_{sw} (em cm²/m)

Fim do programa.

5.3 – Programa em Fortran 90

A listagem a seguir apresenta o código do programa em Fortran 90. Os dados são lidos do arquivo DADOS.TXT. Para facilitar a comparação com os programas em Visual Basic e Visual C#, cada dado é lido em uma linha do arquivo. As mensagens de alerta e os resultados são mostrados na tela. Os dados devem ser fornecidos usando somente o ponto como separador decimal. O arquivo com a listagem do código-fonte em Fortran 90 pode ser obtido em www.editoradunas.com.br/programas.htm.

LISTAGEM DO CÓDIGO-FONTE EM FORTRAN 90

```
!
!Dimensionamento ao Esforço Cortante
!
!PROGRAMA PRINCIPAL
!
!Declarando variáveis reais de dupla precisão
IMPLICIT REAL*8(A-H,O-Z)
!
!Abertura do arquivo de dados
OPEN(1,FILE='DADOS.TXT')
```

```
!  
!LEITURA DOS DADOS  
!  
!Os dados são lidos do arquivo DADOS.TXT  
!Cada dado deve estar em uma linha do arquivo  
!  
!  
!  
!ENTRADA DE DADOS  
!  
!  
!Resistência característica à compressão do concreto em MPa  
read(1,*)fck  
!  
!Tensão de escoamento característica do aço em MPa  
read(1,*)fyk  
!  
!Coeficientes parciais de segurança:  
!para o concreto  
read(1,*)gamac  
!  
!para o aço  
read(1,*)gamas  
!  
!para o esforço cortante  
read(1,*)gamaf  
!  
!Largura da seção: bw em cm  
read(1,*)bw  
!  
!Altura útil em cm  
read(1,*)d  
!  
!Esforço cortante de serviço em kN  
read(1,*)vk  
!  
!  
!FIM DA ENTRADA DE DADOS  
!  
!INÍCIO DOS CÁLCULOS  
!  
!Resistências e cortante de cálculo  
!  
fcd=fck/gamac  
fyd=fyk/gamas  
vd=gamaf*vk  
!  
!Tensão convencional de cisalhamento  
!  
twd=vd/(bw*d)  
!Passando para MPa  
twd=10*twd  
!  
!Tensão de cisalhamento última  
av=1-fck/250  
twu=0.27*av*fcd  
!
```

```

!Verificação do esmagamento das bielas
If(twd>twu)Then
! Está ocorrendo ruptura das bielas
! Colocar mensagem para o usuário aumentar as dimensões da seção transversal e
! parar o processamento
!
  write(*,10)
10 format(//,2x,'Esmagamento da biela de compressão. Aumente as dimensões da seção transversal.')
  goto 100
End If
!
!Tensão Talc de redução da armadura
!Se gamac=1.4 irão resultar as expressões (7.4.6) e (7.4.7) do Volume 1 de Curso
!de Concreto Armado
!
If(fck<=50) Then
  a=2./3.
  tc=0.126*(fck**a)/gamac
Else
  tc=0.8904*Log(1+0.11*fck)/gamac
End If
!
!Tensão Tald para cálculo da armadura
!
td=1.11*(twd-tc)
If(td<0)Then
  td=0
End If
!
!Limitação da tensão de escoamento do aço conforme a NBR-6118
!
If(fyd>435)Then
  fyd=435
End If
!
!Cálculo da armadura
asw=100*bw*td/fyd
!
!Cálculo da armadura mínima:
!A tensão fyk deve ser menor ou igual a 500 MPa
fykmax=fyk
If(fykmax>500)Then
  fykmax= 500
End If
!
!Resistência média à tração do concreto
!
If(fck<=50)Then
  a=2./3.
  fctm=0.3*(fck**a)
Else
  fctm=2.12*Log(1+0.11*fck)
End If
!
!Taxa mínima de armadura (ver equação (7.4.13) do Volume 1)
romin=0.2*fctm/fykmax
aswmin=romin*100*bw

```

```

!
!Verificação
!
If(asw<aswmin)Then
    asw=aswmin
End If
!
!MOSTRAR O RESULTADO NA TELA
!
write(*,20)asw
20 format(//,5x,'Area de estribos (cm2/m)='f6.2)
100 aux=0 !terminando o programa
End

```

5.4 – Programa em Visual Basic 2017

Na fig. 5.4.1 apresenta-se o formulário do programa escrito em Visual Basic 2017. Os dados podem ser fornecidos usando o ponto ou a vírgula como separador decimal.

The form is titled "Dimensionamento ao Esforço Cortante". It features a diagram of a rectangular and a T-shaped cross-section on the left, both with width b_w and effective depth d . The input fields are organized as follows:

- Materiais:** f_{ck} (MPa) and f_{yk} (MPa).
- Coefficientes parciais:** γ_{mac} , γ_{mas} , and γ_{maf} .
- Dimensões da seção transversal:** b_w (cm) and d (cm).
- Esforço cortante de serviço:** V_k (kN).
- Armadura:** A_{sw} (cm²/m).

A "CALCULAR" button is located at the bottom right of the form.

Fig. 5.4.1 – Formulário do programa em Visual Basic 2017

O formulário contém os seguintes controles:

Tipo	Nome	Descrição
PictureBox	PictureBox1	Imagem das seções
Label	Label1	f_{ck} (MPa) =
TextBox	TextBox1	Valor de f_{ck}
Label	Label2	f_{yk} (MPa) =
TextBox	TextBox2	Valor de f_{yk}
Label	Label3	γ_{mac} =
TextBox	TextBox3	Valor de γ_{mac}
Label	Label4	γ_{mas} =
TextBox	TextBox4	Valor de γ_{mas}
Label	Label5	γ_{maf} =
TextBox	TextBox5	Valor de γ_{maf}
Label	Label6	b_w (cm)=

TextBox	TextBox6	Valor de bw
Label	Label7	d (cm)=
TextBox	TextBox7	Valor de d
Label	Label8	Vk (kN)=
TextBox	TextBox8	Valor de Vk
Label	Label9	Asw (cm ² /m)=
TextBox	TextBox9	Valor de Asw
Button	Button1	CALCULAR
GroupBox	GroupBox1	Materiais
GroupBox	GroupBox2	Coeficientes parciais
GroupBox	GroupBox3	Dimensões da seção transversal
GroupBox	GroupBox4	Esforço cortante de serviço
GroupBox	GroupBox5	Armadura

Para criar o projeto, procede-se como descrito na seção 1.4 do capítulo 1. Utilizando a Caixa de Ferramentas, são introduzidos os controles no Form1. A propriedade Text do Form1 é “Dimensionamento ao Esforço Cortante”, como aparece no topo do formulário.

O arquivo com a listagem do código-fonte em Visual Basic 2017 pode ser obtido em www.editoradunas.com.br/programas.htm. Após a elaboração do formulário, basta clicar com o botão direito do mouse sobre o formulário para Exibir Código e colar o código-fonte.

LISTAGEM DO CÓDIGO-FONTE EM VISUAL BASIC 2017

Public Class Form1

Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click

Dim fck, fyk, gamac, gamas, gamaf, vk As Double

Dim bw, d, fcd, fyd, vd, twd, av, twu As Double

Dim a, tc, td, asw, fykmax As Double

Dim fctm, romin, aswmin As Double

,

,

'Esta sub-rotina corresponde ao botão CALCULAR.

,

,

,

'Dimensionamento ao esforço cortante

,

'ENTRADA DE DADOS

,

'Os dados são lidos das caixas de texto do formulário

,

'Resistência característica à compressão do concreto em MPa

"Substituindo a vírgula por ponto

TextBox1.Text = Replace(TextBox1.Text, ",", ".")

fck = Val(TextBox1.Text)

,

'Tensão de escoamento característica do aço em MPa

TextBox2.Text = Replace(TextBox2.Text, ",", ".")

fyk = Val(TextBox2.Text)

```

'
'Coeficientes parciais de segurança:
'para o concreto
TextBox3.Text = Replace(TextBox3.Text, ",", ".")
gamac = Val(TextBox3.Text)
'
'para o aço
TextBox4.Text = Replace(TextBox4.Text, ",", ".")
gamas = Val(TextBox4.Text)
'
'para o esforço cortante
TextBox5.Text = Replace(TextBox5.Text, ",", ".")
gamaf = Val(TextBox5.Text)
'
'Largura da seção: bw em cm
TextBox6.Text = Replace(TextBox6.Text, ",", ".")
bw = Val(TextBox6.Text)
'
'Altura útil em cm
TextBox7.Text = Replace(TextBox7.Text, ",", ".")
d = Val(TextBox7.Text)
'
'Esforço cortante de serviço em kN
TextBox8.Text = Replace(TextBox8.Text, ",", ".")
vk = Val(TextBox8.Text)
'
'
'FIM DA ENTRADA DE DADOS
'
'INÍCIO DOS CÁLCULOS
'
'Resistências e cortante de cálculo
'
fcd = fck / gamac
fyd = fyk / gamas
vd = gamaf * vk
'
'Tensão convencional de cisalhamento
'
twd = vd / (bw * d)
'Passando para MPa
twd = 10 * twd
'
'Tensão de cisalhamento última
av = 1 - fck / 250
twu = 0.27 * av * fcd
'
'Verificação do esmagamento das bielas
If (twd > twu) Then
    ' Está ocorrendo ruptura das bielas
    ' Colocar mensagem para o usuário aumentar as dimensões da seção transversal e parar o
    processamento
    '
    MsgBox("Esmagamento da biela de compressão. Aumente as dimensões da seção
    transversal.", vbOKOnly + vbExclamation)
    Exit Sub
End If

```

```

',
'Tensão Talc de redução da armadura
'Se gamac=1.4 irão resultar as expressões (7.4.6) e (7.4.7) do Volume 1 de Curso de Concreto
Armado
',
If (fck <= 50) Then
  a = 2 / 3
  tc = 0.126 * (fck ^ a) / gamac
Else
  tc = 0.8904 * Math.Log(1 + 0.11 * fck) / gamac
End If
',
'Tensão Tald para cálculo da armadura
',
td = 1.11 * (twd - tc)
If (td < 0) Then
  td = 0
End If
',
'Limitação da tensão de escoamento do aço conforme a NBR-6118
',
If (fyd > 435) Then
  fyd = 435
End If
',
'Cálculo da armadura
asw = 100 * bw * td / fyd
',
'Cálculo da armadura mínima:
'A tensão fyk deve ser menor ou igual a 500 MPa
fykmax = fyk
If (fykmax > 500) Then
  fykmax = 500
End If
',
'Resistência média à tração do concreto
',
If (fck <= 50) Then
  a = 2 / 3
  fctm = 0.3 * (fck ^ a)
Else
  fctm = 2.12 * Math.Log(1 + 0.11 * fck)
End If
',
'Taxa mínima de armadura (ver equação (7.4.13) do Volume 1)
romin = 0.2 * fctm / fykmax
aswmin = romin * 100 * bw
',
'Verificação
',
If (asw < aswmin) Then
  asw = aswmin
End If
',
'Convertendo a saída para duas casas decimais
asw = FormatNumber(asw, 2)
',

```



```
TextBox9.Text = asw
```

```
End Sub
```

```
End Class
```

5.5 - Programa em Visual C# 2017

Na fig. 5.5.1 apresenta-se o formulário do programa escrito em Visual C# 2017. O formulário é inteiramente análogo ao do programa em Visual Basic 2017. Os dados podem ser fornecidos usando o ponto ou a vírgula como separador decimal. Os controles são os mesmos do programa em Visual Basic.

O arquivo com a listagem do código-fonte em Visual C# 2017 pode ser obtido em www.editoradunas.com.br/programas.htm. Após a elaboração do formulário, basta clicar com o botão direito do mouse sobre o formulário para Exibir Código e colar o código-fonte.

The screenshot shows a Windows application window titled "Dimensionamento ao Esforço Cortante". On the left, there is a diagram of a T-beam cross-section with labels for width b_w and effective depth d . The main area contains several input groups:

- Materiais:** f_{ck} (MPa) and f_{yk} (MPa).
- Coefficientes parciais:** γ_{mac} , γ_{mas} , and γ_{maf} .
- Dimensões da seção transversal:** b_w (cm) and d (cm).
- Esforço cortante de serviço:** V_k (kN).
- Armadura:** A_{sw} (cm²/m).

 A "CALCULAR" button is located at the bottom right of the form.

Fig. 5.5.1 – Formulário do programa em Visual C# 2017

LISTAGEM DO CÓDIGO-FONTE EM VISUAL C# 2017

```
using System;
using System.Globalization;
using System.Windows.Forms;

namespace CORTANTE
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
        }
    }
}
```

```
private void button1_Click(object sender, EventArgs e)
{
    double fck, fyk, gamac, gamas, gamaf, vk;
    double bw,d, fcd, fyd, vd, twd, av, twu;
    double a, tc, td, asw, fykmax;
    double fctm, romin, aswmin;
    //
    //
    //Esta sub-rotina corresponde ao botão CALCULAR.
    //
    //
    //
    //Dimensionamento ao esforço cortante
    //
    //ENTRADA DE DADOS
    //
    //Os dados são lidos das caixas de texto do formulário
    //
    //Resistência característica à compressão do concreto em MPa
    //Substituindo a vírgula por ponto
    textBox1.Text = textBox1.Text.Replace(",", ".");
    fck = Convert.ToDouble(textBox1.Text, CultureInfo.InvariantCulture);
    //
    //Tensão de escoamento característica do aço em MPa
    textBox2.Text = textBox2.Text.Replace(",", ".");
    fyk = Convert.ToDouble(textBox2.Text, CultureInfo.InvariantCulture);
    //
    //Coeficientes parciais de segurança:
    //para o concreto
    textBox3.Text = textBox3.Text.Replace(",", ".");
    gamac = Convert.ToDouble(textBox3.Text, CultureInfo.InvariantCulture);
    //
    //para o aço
    textBox4.Text = textBox4.Text.Replace(",", ".");
    gamas = Convert.ToDouble(textBox4.Text, CultureInfo.InvariantCulture);
    //
    //para o esforço cortante
    textBox5.Text = textBox5.Text.Replace(",", ".");
    gamaf = Convert.ToDouble(textBox5.Text, CultureInfo.InvariantCulture);
    //
    //Largura da seção: bw em cm
    textBox6.Text = textBox6.Text.Replace(",", ".");
    bw = Convert.ToDouble(textBox6.Text, CultureInfo.InvariantCulture);
    //
    //Altura útil em cm
    textBox7.Text = textBox7.Text.Replace(",", ".");
    d = Convert.ToDouble(textBox7.Text, CultureInfo.InvariantCulture);
    //
    //Esforço cortante de serviço em kN
    textBox8.Text = textBox8.Text.Replace(",", ".");
    vk = Convert.ToDouble(textBox8.Text, CultureInfo.InvariantCulture);
    //
    //
    //FIM DA ENTRADA DE DADOS
    //
    //INÍCIO DOS CÁLCULOS
```

```

//
//Resistências e cortante de cálculo
//
fcd = fck / gamac;
fyd = fyk / gamas;
vd = gamaf * vk;
//
//Tensão convencional de cisalhamento
//
twd = vd / (bw * d);
//Passando para MPa
twd = 10 * twd;
//
//Tensão de cisalhamento última
av = 1 - fck / 250;
twu = 0.27 * av * fcd;
//
//Verificação do esmagamento das bielas
if(twd > twu)
{
    // Está ocorrendo ruptura das bielas
    // Colocar mensagem para o usuário aumentar as dimensões da seção transversal e parar o
processamento
    //
    MessageBox.Show("Esmagamento da biela de compressão. Aumente as dimensões da seção
transversal.");
    return;
}
//
//Tensão Talc de redução da armadura
//Se gamac=1.4 irão resultar as expressões (7.4.6)e (7.4.7) do Volume 1 de Curso de Concreto
Armado
//
if(fck <= 50)
{
    a = 2.0 / 3.0;
    tc = 0.126 * Math.Pow(fck, a) / gamac;
}
else
{
    tc = 0.8904 * Math.Log(1 + 0.11 * fck) / gamac;
}
//
//Tensão Tald para cálculo da armadura
//
td = 1.11 * (twd - tc);
if(td < 0)
{
    td = 0;
}
//
//Limitação da tensão de escoamento do aço conforme a NBR-6118
//
if(fyd > 435)
{
    fyd = 435;
}

```

```

//
//Cálculo da armadura
asw = 100 * bw * td / fyd;
//
//Cálculo da armadura mínima:
//A tensão fyk deve ser menor ou igual a 500 MPa
fykmax = fyk;
if(fykmax > 500)
{
    fykmax = 500;
}
//
//Resistência média à tração do concreto
//
if(fck <= 50)
{
    a = 2.0 / 3.0;
    fctm = 0.3 * Math.Pow(fck, a);
}
else
{
    fctm = 2.12 * Math.Log(1 + 0.11 * fck);
}
//
//Taxa mínima de armadura (ver equação (7.4.13) do Volume 1)
romin = 0.2 * fctm / fykmax;
aswmin = romin * 100 * bw;
//
//Verificação
//
if(asw < aswmin)
{
    asw = aswmin;
}
//Convertendo a saída para duas casas decimais
//
decimal saida = Decimal.Round(Convert.ToDecimal(asw), 2);
//
//MOSTRAR O RESULTADO
//
textBox9.Text = Convert.ToString(saida);
}
}
}

```

5.6 – Exemplos para testar os programas

Os resultados apresentados a seguir servem para o leitor testar os programas. Os exemplos correspondem à seção retangular da fig. 5.6.1. Esses exemplos foram resolvidos manualmente no Capítulo 7 do Volume 1.

A tensão de escoamento característica do aço é $f_{yk} = 600$ MPa.

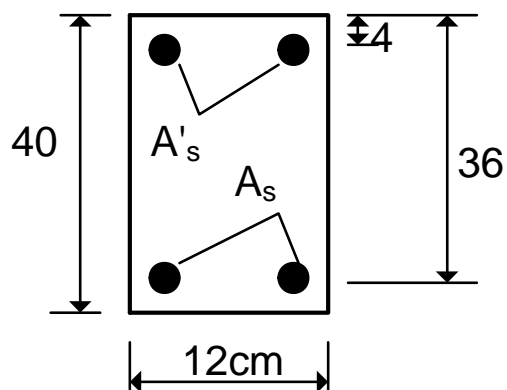


Fig. 5.6.1 - Seção transversal

Na tabela 5.6.1 apresentam-se alguns resultados obtidos com os programas.

Tabela 5.6.1 – Armadura transversal A_{sw} cm²/m

Exemplo	V_k (kN)	$f_{ck} = 20$ MPa	$f_{ck} = 30$ MPa	$f_{ck} = 40$ MPa
1	30	1,06 *	1,39 *	1,68
2	40	1,94	1,39 *	1,68
3	50	2,93	2,30	1,74
4	60	3,92	3,29	2,73
* Os valores da tabela 7.4.2 do Volume 1 são arredondados.				

Capítulo 6

FLEXO-COMPRESSÃO NORMAL Dimensionamento de Seções Retangulares

6.1 – Apresentação do problema

O objetivo é dimensionar uma seção retangular com várias camadas de armadura, submetida à flexo-compressão normal. A geometria da seção é indicada na fig. 6.1.1. A teoria é apresentada no Capítulo 2 do Volume 3.

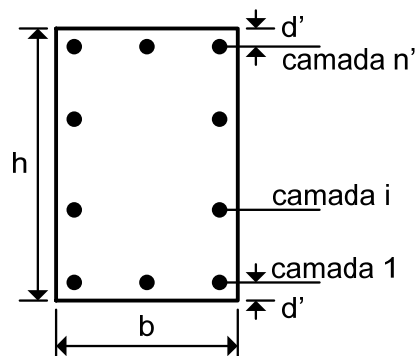


Fig. 6.1.1 - Seção retangular com várias camadas de armadura

Os dados do problema são os seguintes:

- a) propriedades dos materiais: f_{ck} , f_{yk} , E_s
- b) coeficientes parciais de segurança: γ_c , γ_s , γ_f
- c) dimensões da seção transversal: b , h , d'
- d) número de camadas de armadura: $n' \geq 2$
- e) número de barras de cada camada: ≥ 2
- f) esforço normal e momento fletor de serviço: N_k , M_k

O valor requerido é a área total de aço, A_s , na seção transversal.

6.2 – Roteiro para programação

1) Entrada de dados

Ler as propriedades dos materiais: f_{ck} (MPa), f_{yk} (MPa), E_s (GPa)

Ler os coeficientes parciais de segurança: γ_c , γ_s , γ_f

Ler as dimensões da seção (em cm): b , h , d'

Ler o número de camadas de armadura: $n' \geq 2$

Ler o número de barras de cada camada: $n_i(i) \geq 2$

Ler o esforço normal de serviço (em kN): N_k

Ler o momento fletor de serviço (em kNm): M_k

Fim da entrada de dados e início dos cálculos.

2) Parâmetros do diagrama retangular para o concreto

- Se $f_{ck} \leq 50$ MPa:

$$\lambda = 0,8$$

$$\alpha_c = 0,85$$

$$\varepsilon_u \left(\frac{o}{oo} \right) = 3,5$$

$$\varepsilon_o \left(\frac{o}{oo} \right) = 2$$

- Se $f_{ck} > 50$ MPa:

$$\lambda = 0,8 - \frac{(f_{ck} - 50)}{400}$$

$$\alpha_c = 0,85 \left[1 - \frac{(f_{ck} - 50)}{200} \right]$$

$$\varepsilon_u \left(\frac{o}{oo} \right) = 2,6 + 35 \left(\frac{90 - f_{ck}}{100} \right)^4$$

$$\varepsilon_o \left(\frac{o}{oo} \right) = 2,0 + 0,085(f_{ck} - 50)^{0,53}$$

3) Parâmetro kapa

$$\kappa = 1 - \varepsilon_o / \varepsilon_u$$

4) Conversão de unidades para kN e cm

$$f_{ck} = f_{ck} / 10$$

$$f_{yk} = f_{yk} / 10$$

$$E_s = 100 E_s$$

$$M_k = 100 M_k$$

5) Resistências e esforços de cálculo

$$f_{cd} = f_{ck} / \gamma_c$$

$$\sigma_{cd} = \alpha_c f_{cd}$$

$$f_{yd} = f_{yk} / \gamma_s$$

$$N_d = \gamma_f N_k$$

$$M_d = \gamma_f M_k$$

6) Cálculo do número total de barras na seção transversal

$$n = \sum_{i=1}^{n'} ni(i)$$

7) Parâmetro geométrico e área da seção transversal

$$\delta = d'/h$$

$$A_c = bh$$

8) Esforços solicitantes reduzidos

$$\nu = \frac{N_d}{A_c \sigma_{cd}}$$

$$\mu = \frac{M_d}{A_c h \sigma_{cd}}$$

9) Caso particular de compressão centrada

- Se $\mu = 0$:

$$\varepsilon_s = \varepsilon_o \text{ (deformação uniforme nas armaduras)}$$

Chamar uma sub-rotina para calcular a tensão σ_{sdo} .

$$\omega = \frac{(\nu - 1)f_{yd}}{\sigma_{sdo}}$$

$$\text{Se } \omega < 0 \rightarrow \omega = 0$$

Ir para o passo 14) no final do programa para calcular A_s .

10) Montagem do vetor β

Ver equação (2.2.5) do Capítulo 2 do Volume 3.

$$\beta_i = \delta + \frac{(n' - i)(1 - 2\delta)}{(n' - 1)}, \text{ com } i = 1, \dots, n'$$

11) Determinação do intervalo solução para a linha neutra

$$\text{Profundidade inicial: } \xi_i = 0$$

Chamar sub-rotina para calcular a função $f_i = f(\xi_i)$.

$$\text{Profundidade final: } \xi_f = 1000$$

Chamar sub-rotina para calcular a função $f_f = f(\xi_f)$.

Calcular $prod = f_i f_f$

➤ Fazer enquanto $prod > 0$:

$$\xi_i = \xi_f$$

$$f_i = f_f$$

$$\xi_f = 10\xi_f$$

Chamar sub-rotina para calcular a função $f_f = f(\xi_f)$.

$$prod = f_i f_f$$

⬆ Retornar

A linha neutra está no intervalo $[\xi_i, \xi_f]$.

12) Processo iterativo da bissecante para encontrar a linha neutra

Iterar até a convergência (processo da bissecante):

Fazer: $f_k = 1$

➤ Fazer enquanto $|f_k| > 0.001$:

$$\xi_k = (\xi_i f_f - \xi_f f_i) / (f_f - f_i)$$

Chamar sub-rotina para calcular a função $f_k = f(\xi_k)$.

Calcular $prod = f_k f_i$

• Se $prod \geq 0$

$$\xi_i = \xi_k$$

$$f_i = f_k$$

• Se $prod < 0$

$$\xi_f = \xi_k$$

$$f_f = f_k$$

⬆ Retornar

13) Cálculo da taxa de armadura

Emprego das equações (2.5.9) e (2.5.10) do Capítulo 2 do Volume 3: será usada aquela que tiver o maior denominador, denominados de *soma1* e *soma2*, respectivamente.

• Se $|soma1| \geq |soma2|$:

$$\omega = \frac{n f_{yd} (v - r_c)}{soma1} = \frac{n'}{\sum_{i=1}^{n'} n_i \sigma_{sdi}}$$

- Se $|soma1| < |soma2|$:

$$\omega = \frac{nf_{yd}(0,5\nu - \mu - r_c\beta_c)}{soma2 = \sum_{i=1}^{n'} n_i\beta_i\sigma_{sdi}}$$

$$\text{Se } \omega < 0 \rightarrow \omega = 0.$$

14) Área de aço

$$A_s = \omega A_c \frac{\sigma_{cd}}{f_{yd}}$$

15) Armadura mínima

$$\nu_o = \frac{N_d}{A_c f_{cd}}$$

$$\rho_{\min} = 0,15 \frac{f_{cd}}{f_{yd}} \nu_o \geq 0,40\%$$

$$A_{s,\min} = \rho_{\min} A_c$$

$$\text{Se } A_s < A_{s,\min} \rightarrow A_s = A_{s,\min}$$

Imprimir ou mostrar resultados: A_s , $A_{s,\min}$ (em cm^2)

Fim do programa principal.

B) Sub-rotina função

Essa sub-rotina calcula o valor da função $f(\xi)$ dada na equação (2.5.11) do Volume 3. A profundidade relativa da linha neutra, $\xi = x/h$, é um parâmetro de entrada na sub-rotina.

1) Constantes para cálculo das deformações na armadura

$$\xi_1 = \left(\frac{\varepsilon_u}{\varepsilon_u + 10\text{‰}} \right) \beta_1$$

- Se $\xi \leq \xi_1$, a linha neutra está no domínio 2:

$$c = 0,01/(\beta_1 - \xi)$$
- Se $\xi_1 < \xi \leq 1$, a linha neutra está nos domínios 3, 4 ou 4a:

$$c = (\varepsilon_u / 1000) / \xi$$
- Se $\xi > 1$, a linha neutra está no domínio 5:

$$c = (\varepsilon_o / 1000) / (\xi - \kappa)$$

2) Resultante de compressão no concreto

- Se $\xi < 1/\lambda$:

$$r_c = \lambda \xi$$

$$\beta_c = 0,5 \lambda \xi$$

- Se $\xi \geq 1/\lambda$:

$$r_c = 1$$

$$\beta_c = 0,5$$

3) Cálculo dos somatórios das equações (2.5.9) e (2.5.10)

$$soma1 = 0$$

$$soma2 = 0$$

➡ Somar de 1 a n': $\sum_{i=1}^{n'}$

$$\varepsilon_{si} = c(\xi - \beta_i)$$

Chamar sub-rotina para calcular a tensão σ_{si} .

$$soma1 = soma1 + ni(i)\sigma_{si}$$

$$\blacktriangleleft soma2 = soma2 + ni(i)\beta_i\sigma_{si}$$

4) Função $f(\xi)$

$$f(\xi) = (\mu - 0,5\nu + r_c\beta_c)soma1 + (\nu - r_c)soma2$$

Fim da sub-rotina.

C) Sub-rotina tensão

Parâmetros de entrada: E_s , ε_s , f_{yd}

Parâmetro de saída: σ_{sd}

1) Trabalhar com o valor absoluto da deformação

$$\varepsilon_{ss} = |\varepsilon_s|$$

2) Deformação de escoamento de cálculo do aço

$$\varepsilon_{yd} = f_{yd} / E_s$$

3) Cálculo da tensão

- Se $\varepsilon_{ss} < \varepsilon_{yd}$:

$$\sigma_{sd} = E_s \varepsilon_{ss}$$

- Se $\varepsilon_{ss} \geq \varepsilon_{yd}$:

$$\sigma_{sd} = f_{yd}$$

4) Acertando o sinal da tensão

Se $\varepsilon_s < 0$, fazer $\sigma_{sd} = -\sigma_{sd}$

Fim da sub-rotina.

6.3– Programa em Fortran 90

A listagem a seguir apresenta o código do programa em Fortran 90. Os dados são lidos do arquivo DADOS.TXT. Para facilitar a comparação com os programas em Visual Basic e Visual C#, cada dado é lido em uma linha do arquivo. As mensagens de alerta e os resultados são mostrados na tela. Os dados devem ser fornecidos usando somente o ponto como separador decimal. O arquivo com a listagem do código-fonte em Fortran 90 pode ser obtido em www.editoradunas.com.br/programas.htm.

LISTAGEM DO CÓDIGO-FONTE EM FORTRAN 90

```
!
!Dimensionamento à Flexo-Compressão Normal
!Seções retangulares com várias camadas de armadura
!O programa está limitado a 20 camadas
!
!PROGRAMA PRINCIPAL
!
!Declarando variáveis reais de dupla precisão
IMPLICIT REAL*8(A-H,O-Z)
!Variáveis públicas
COMMON /A1/ ni(20),beta(20)
COMMON /A2/ n,nl
COMMON /A3/ delta,ani,ami,fyd,fcd,es
COMMON /A4/ alfac,alamb,e0,eu,akapa
!
!
!Abertura do arquivo de dados
OPEN(1,FILE='DADOS.TXT')
!
!LEITURA DOS DADOS
!
!Os dados são lidos do arquivo DADOS.TXT
!Cada dado deve estar em uma linha do arquivo

!Resistência característica à compressão do concreto em MPa
read(1,*)fck
!
!Tensão de escoamento característica do aço em MPa
read(1,*)fyk
!
```

```

!Módulo de elasticidade do aço em GPa
read(1,*)es
!
!Coeficientes parciais de segurança:
!para o concreto
read(1,*)gamac
!para o aço
read(1,*)gamas
!para os esforços solicitantes
read(1,*)gamaf
!
!Dados da seção transversal:
!Largura em cm
read(1,*)b
!Altura em cm
read(1,*)h
!Distância d!
read(1,*)dl
!
!Número de camadas de armadura:
!A seção deve ter, no mínimo, duas camadas de armadura
!nl>=2
read(1,*)nl
!
!Número de barras das camadas de armadura:
!As camadas são numeradas de baixo para cima
!Cada camada deve ter, no mínimo, duas barras
do i=1,nl
    read(1,*)ni(i)
end do
!Esforços solicitantes de serviço:
!Entrar com valores positivos
!Esforço normal (em kN)
read(1,*)ank
!Momento fletor (em kNm)
read(1,*)amk
!
!
!FIM DA ENTRADA DE DADOS
!
!INÍCIO DOS CÁLCULOS
!
!Parâmetros do diagrama retangular
If(fck<=50)Then
    alamb=0.8
    alfac=0.85
    eu=3.5
    e0=2
Else
    alamb=0.8-(fck-50)/400
    alfac=0.85*(1-(fck-50)/200)
    eu=2.6+35*((90-fck)/100)**4
    e0=2+0.085*((fck-50)**0.53)
End If
!Parâmetro kapa que define o ponto com deformação igual a eo no domínio 5
akapa=1-e0/eu
!

```

```

!Conversão de unidades: transformando para kN e cm
fck=fck/10
fyk=fyk/10
es=100*es
amk=100*amk
!
!Resistências de cálculo
fcd=fck/gamac
tcd=alfac*fcd
fyd=fyk/gamas
!
!Esforços solicitantes de cálculo
aand=gamaf*ank
amd=gamaf*amk
!
!Cálculo do número total de barras na seção
n=0
do i=1,nl
    n=n+ni(i)
end do
!
!Parâmetro geométrico
delta=dl/h
!
!Área da seção de concreto
ac=b*h
!
!Esforços reduzidos
ani=aand/(ac*tcd)
ami=amd/(ac*h*tcd)
!
!Caso particular de compressão centrada
!
If(ami.Eq.0)Then
    esi=e0/1000
    Call Tensao(esi,tsd0)
    w=(ani-1)*fyd/tsd0
    If(w<0)Then
        w=0
    End If
    GoTo 100
End If
!
!Flexo-compressão normal
!
!Montagem do vetor beta
!Ver equação (2.2.5) do Volume 3 de Curso de Concreto Armado
!
do i=1,nl
    beta(i)=delta+(nl-i)*(1-2*delta)/(nl-1)
end do
!Processo iterativo da bissecante
!
!Determinação do intervalo solução
!
!Valor inicial para a linha neutra adimensional qsi=x/h
qi=0

```

```

!Chamar sub-rotina para calcular o valor da função  $f_i=f(q_i)$ 
Call Funcao(qi,rc,bc,soma1,soma2,fi)
!
!Valor final para a linha neutra adimensional  $q_{si}=x/h$ 
qf=1000
!Chamar sub-rotina para calcular o valor da função  $f_f=f(q_f)$ 
Call Funcao(qf,rc,bc,soma1,soma2,ff)
!
prod=fi*ff
!Modificando os extremos do intervalo solução até que  $prod \leq 0$ 
Do While (prod>0)
    qi=qf
    fi=ff
    qf=10*qf
    Call Funcao(qf,rc,bc,soma1,soma2,ff)
    prod=fi*ff
End do
!
!O intervalo solução foi definido
!A linha neutra  $q_{si}$  fica entre  $[q_i, q_f]$ 
!
!Processo iterativo da bissecante
fk=1
Do While (Abs(fk)>0.001)
    qk=(qi*ff-qf*fi)/(ff-fi)
    Call Funcao(qk,rc,bc,soma1,soma2,fk)
    prod=fk*fi
    If(prod>=0)Then
        qi=qk
        fi=fk
    Else
        qf=qk
        ff=fk
    End If
End do
!Convergência alcançada
!qk é a raiz da função  $f(q_{si})$  dada na equação (2.5.11) do Volume 3 de Curso de
!Concreto Armado
!
!Cálculo da taxa mecânica de armadura
If(Abs(soma1)>=Abs(soma2))Then
    ! Uso da equação (2.5.9) do Volume 3
    w=n*fyd*(ani-rc)/soma1
Else
    ! Uso da equação (2.5.10) do Volume 3
    w=n*fyd*(0.5*ani-ami-rc*bc)/soma2
End If
If(w<0)Then
    w=0
End If
!
!Cálculo da área de armadura total
100 aas=w*ac*tcd/fyd
!
!Armadura mínima da NBR-6118 para pilares
!
ani0=aand/(ac*fcd)

```

```

romin=0.15*fcd*ani0/fyd
If(romin<0.004)Then
  romin=0.004
End If
asmin=romin*ac
!
!Armadura a ser adotada
ase=aas
If(ase<asmin) Then
  ase=asmin
End If
!
!MOSTRAR O RESULTADO
!
write(*,10)aas
10 format(/,5x,'Armadura calculada: As(cm2)='f6.2)
write(*,20)asmin
20 format(/,5x,'Armadura minima: As,min(cm2)='f6.2)
write(*,30)ase
30 format(/,5x,'Armadura a ser adotada: Ase(cm2)='f6.2)
!
End
!
!
!
Subroutine Funcao(qsi,rc,bc,soma1,soma2,f)
IMPLICIT REAL*8(A-H,O-Z)
!Variáveis públicas
COMMON /A1/ ni(20),beta(20)
COMMON /A2/ n,nl
COMMON /A3/ delta,ani,ami,fyd,fcd,es
COMMON /A4/ alfac,alamb,e0,eu,akapa
!
!Calcula o valor da função f(qsi) dada na equação (2.5.11) do Volume 3 de Curso de
!Concreto Armado
!
!qsi=x/h é a profundidade relativa da linha neutra
!rc é a resultante de compressão do concreto adimensional dada na equação (2.4.4)
!bc é a posição da resultante adimensional dada na equação (2.4.5)
!soma1 é o somatório contido no denominador da equação (2.5.9)
!soma2 é o somatório contido no denominador da equação (2.5.10)
!f é o resultado da equação (2.5.11)
!
!O único parâmetro de entrada é qsi, pois as demais variáveis são públicas.
!Os parâmetros rc,bc,soma1,soma2,f são calculados nessa sub-rotina
!
!
!Constantes para o cálculo das deformações das camadas da armadura
ql=eu*beta(1)/(eu+10)
If(qsi<=ql)Then
  !A linha neutra está no domínio 2
  c=0.01/(beta(1)-qsi)
ElseIf(qsi<=1)Then
  !A linha neutra está nos domínios 3,4 e 4a
  c=eu/(1000*qsi)
Else
  !A linha neutra está no domínio 5

```



```

    c=(e0/1000)/(qsi-akapa)
End If
!
!Resultante de compressão no concreto
If(qsi<1/alamb)Then
    rc=alamb*qsi
    bc=0.5*alamb*qsi
Else
    rc=1
    bc=0.5
End If
soma1=0
soma2=0
do i=1,nl
    esi=c*(qsi-beta(i))
    Call Tensao(esi,tsi)
    soma1=soma1+ni(i)*tsi
    soma2=soma2+ni(i)*beta(i)*tsi
end do
!Função f(qsi)
f=(ami-0.5*ani+rc*bc)*soma1+(ani-rc)*soma2

Return
End
!
!
!
Subroutine Tensao(esl,tsl)
IMPLICIT REAL*8(A-H,O-Z)
!Variáveis públicas
COMMON /A3/ delta,ani,ami,fyd,fcd,es
!
!Calcula a tensão no aço
!es = módulo de elasticidade do aço em kN/cm2
!esl = deformação de entrada
!fyd = tensão de escoamento de cálculo em kN/cm2
!tsl = tensão de saída em kN/cm2
!
!Trabalhando com deformação positiva
ess=Abs(esl)
eyd=fyd/es
If(ess<eyd)Then
    tsl=es*ess
Else
    tsl=fyd
End If
!Trocando o sinal se necessário
If(esl<0)Then
    tsl=-tsl
End If
Return
End

```

6.4 – Programa em Visual Basic 2017

Na fig. 6.4.1 apresenta-se o formulário do programa escrito em Visual Basic 2017. Os dados podem ser fornecidos usando o ponto ou a vírgula como separador decimal. Os dados das camadas de armadura são fornecidos em uma grade, a qual é exibida assim que o usuário digitar o número de camadas.

The screenshot shows a Windows application window titled "Dimensionamento de Seções Retangulares à Flexo-Compressão Normal". On the left, there is a diagram of a rectangular cross-section with width b and height h . It shows reinforcement layers: "camada 1" at the bottom, "camada i" in the middle, and "camada n'" at the top. Distances d' are indicated from the top and bottom edges to the center of the reinforcement layers. Below the diagram is a table with two columns: "Camada" and "Número de barras". The table is currently empty. To the right of the diagram are several input fields grouped into sections: "Materiais" (with f_{ck} (MPa), f_{yk} (MPa), and E_s (GPa)), "Coeficientes parciais" (with γ_{mac} , γ_{mas} , and γ_{maf}), "Seção transversal" (with b (cm), h (cm), d' (cm), and "Número de camadas n' "), and "Esforços solicitantes de serviço" (with N_k (kN) and M_k (kNm)). At the bottom, there is a "Resposta" section with "Área de aço calculada (cm²)" and "Armadura mínima (cm²)" fields, a "Solução: A_s (cm²)" label, and a "CALCULAR" button.

Fig. 6.4.1 – Formulário do programa em Visual Basic 2017

Para criar o projeto, procede-se como descrito na seção 1.4 do capítulo 1. Utilizando a Caixa de Ferramentas, são introduzidos os controles no Form1. A propriedade Text do Form1 é "Dimensionamento de Seções Retangulares à Flexo-Compressão Normal", como aparece no topo do formulário.

O arquivo com a listagem do código-fonte em Visual Basic 2017 pode ser obtido em www.editoradunas.com.br/programas.htm. Após a elaboração do formulário, basta clicar com o botão direito do mouse sobre o formulário para Exibir Código e colar o código-fonte.

O formulário contém os seguintes controles:

Tipo	Nome	Descrição
PictureBox	PictureBox1	Imagem da seção
Label	Label1	f_{ck} (MPa)=
TextBox	TextBox1	Valor de f_{ck}
Label	Label2	f_{yk} (MPa)=
TextBox	TextBox2	Valor de f_{yk}
Label	Label3	E_s (GPa)=
TextBox	TextBox3	Valor de E_s
Label	Label4	γ_{mac} =
TextBox	TextBox4	Valor de γ_{mac}
Label	Label5	γ_{mas} =
TextBox	TextBox5	Valor de γ_{mas}
Label	Label6	γ_{maf} =

TextBox	TextBox6	Valor de Gamaf
Label	Label7	b (cm)=
TextBox	TextBox7	Valor de b
Label	Label8	h (cm)=
TextBox	TextBox8	Valor de h
Label	Label9	d' (cm)=
TextBox	TextBox9	Valor de d'
Label	Label10	Número de camadas n'=
TextBox	TextBox10	Valor de n'
Label	Label11	Nk (kN)=
TextBox	TextBox11	Valor de Nk
Label	Label12	Mk (kNm)=
TextBox	TextBox12	Valor de Mk
Label	Label13	Área de aço calculada (cm2)=
TextBox	TextBox13	Valor da área de aço calculada
Label	Label14	Armadura mínima (cm2)=
TextBox	TextBox14	Valor da armadura mínima
Label	Label15	Solução: As (cm2)=
TextBox	TextBox15	Valor de As
DataGridView	DataGridView1	Dados das camadas de aço
Button	Button1	CALCULAR
GroupBox	GroupBox1	Materiais
GroupBox	GroupBox2	Coeficientes parciais
GroupBox	GroupBox3	Seção transversal
GroupBox	GroupBox4	Esforços solicitantes de serviço
GroupBox	GroupBox5	Resposta

Para o controle DataGridView1, Habilitar Edição. Para a coluna 1, definir a propriedade ReadOnly como True.

LISTAGEM DO CÓDIGO-FONTE EM VISUAL BASIC 2017

Public Class Form1

Public rc, bc, soma1, soma2, f, tsl As Double

Public es, fyd, beta(20) As Double

Public n, nl, ni(20) As Integer

Public delta, ani, ami, fcd As Double

Public alfac, alamb, e0, eu, akapa As Double

Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click

Dim fck, fyk, gamac, gamas, gamaf As Double

Dim b, h, dl, ank, amk As Double

Dim tcd, aand, amd, ac, w As Double

Dim esi, tsd0, qi, fi, qf, ff, qk, fk, prod As Double

Dim aas, ani0, romin, asmin, ase As Double

,

'Essa sub-rotina corresponde ao botão CALCULAR

,

'Dimensionamento à Flexo-Compressão Normal

'Seções retangulares com várias camadas de armadura

'O programa está limitado a 20 camadas

,

```

,
'ENTRADA DE DADOS
,
'Os dados são lidos das caixas de texto do formulário e da grade.
,
'Resistência característica à compressão do concreto em MPa
'Substituindo a vírgula por ponto
TextBox1.Text = Replace(TextBox1.Text, ",", ".")
fck = Val(TextBox1.Text)
,
'Tensão de escoamento característica do aço em MPa
TextBox2.Text = Replace(TextBox2.Text, ",", ".")
fyk = Val(TextBox2.Text)
,
'Módulo de elasticidade do aço em GPa
TextBox3.Text = Replace(TextBox3.Text, ",", ".")
es = Val(TextBox3.Text)
,
'Coeficientes parciais de segurança:
'para o concreto
TextBox4.Text = Replace(TextBox4.Text, ",", ".")
gamac = Val(TextBox4.Text)
'para o aço
TextBox5.Text = Replace(TextBox5.Text, ",", ".")
gamas = Val(TextBox5.Text)
'para os esforços solicitantes
TextBox6.Text = Replace(TextBox6.Text, ",", ".")
gamaf = Val(TextBox6.Text)
,
'Dados da seção transversal:
'Largura em cm
TextBox7.Text = Replace(TextBox7.Text, ",", ".")
b = Val(TextBox7.Text)
'Altura em cm
TextBox8.Text = Replace(TextBox8.Text, ",", ".")
h = Val(TextBox8.Text)
'Distância d'
TextBox9.Text = Replace(TextBox9.Text, ",", ".")
dl = Val(TextBox9.Text)
,
'Número de camadas de armadura:
'A seção deve ter, no mínimo, duas camadas de armadura
'nl>=2
nl = Val(TextBox10.Text)
,
'Número de barras das camadas de armadura:
'As camadas são numeradas de baixo para cima
'Cada camada deve ter, no mínimo, duas barras
For i = 0 To nl - 1
    'Lendo os dados no DataGridView
    ni(i) = DataGridView1.Item(1, i).Value
Next
'Esforços solicitantes de serviço:
'Entrar com valores positivos
'Esforço normal (em kN)
TextBox11.Text = Replace(TextBox11.Text, ",", ".")
ank = Val(TextBox11.Text)

```

```

'Momento fletor (em kNm)
TextBox12.Text = Replace(TextBox12.Text, ",", ".")
amk = Val(TextBox12.Text)
,
,
'FIM DA ENTRADA DE DADOS
,
'INÍCIO DOS CÁLCULOS
,
'Parâmetros do diagrama retangular
If (fck <= 50) Then
    alamb = 0.8
    alfac = 0.85
    eu = 3.5
    e0 = 2
Else
    alamb = 0.8 - (fck - 50) / 400
    alfac = 0.85 * (1 - (fck - 50) / 200)
    eu = 2.6 + 35 * ((90 - fck) / 100) ^ 4
    e0 = 2 + 0.085 * ((fck - 50) ^ 0.53)
End If
'Parâmetro kapa que define o ponto com deformação igual a eo no domínio 5
akapa = 1 - e0 / eu
,
'Conversão de unidades: transformando para kN e cm
fck = fck / 10
fyk = fyk / 10
es = 100 * es
amk = 100 * amk
,
'Resistências de cálculo
fcd = fck / gamac
tcd = alfac * fcd
fyd = fyk / gamas
,
'Esforços solicitantes de cálculo
aand = gamaf * ank
amd = gamaf * amk
,
'Cálculo do número total de barras na seção
n = 0
For i = 0 To nl - 1
    n = n + ni(i)
Next
,
'Parâmetro geométrico
delta = dl / h
,
'Área da seção de concreto
ac = b * h
,
'Esforços reduzidos
ani = aand / (ac * tcd)
ami = amd / (ac * h * tcd)
,
'Caso particular de compressão centrada
,

```

```

If (ami = 0) Then
    esi = e0 / 1000
    Call Tensao(esi)
    tsd0 = tsl
    w = (ani - 1) * fyd / tsd0
    If (w < 0) Then
        w = 0
    End If
    GoTo 100
End If
'
'Flexo-compressão normal
'
'Montagem do vetor beta
'Ver equação (2.2.5) do Volume 3 de Curso de Concreto Armado
'Aqui a primeira camada tem índice = 0
'A equação foi modificada para compatibilizar
For i = 0 To nl - 1
    beta(i) = delta + (nl - 1 - i) * (1 - 2 * delta) / (nl - 1)
Next
'
'Processo iterativo da bissecante
'
'Determinação do intervalo solução
'
'Valor inicial para a linha neutra adimensional qsi=x/h
qi = 0
'Chamar sub-rotina para calcular o valor da função fi=f(qi)
Call Funcao(qi)
fi = f
'
'Valor final para a linha neutra adimensional qsi=x/h
qf = 1000
'Chamar sub-rotina para calcular o valor da função ff=f(qf)
Call Funcao(qf)
ff = f
prod = fi * ff
'Modificando os extremos do intervalo solução até que prod<=0
Do While (prod > 0)
    qi = qf
    fi = ff
    qf = 10 * qi
    Call Funcao(qf)
    ff = f
    prod = fi * ff
Loop
'
'O intervalo solução foi definido
'A linha neutra qsi fica entre [qi,qf]
'
'Processo iterativo da bissecante
fk = 1
Do While (Math.Abs(fk) > 0.001)
    qk = (qi * ff - qf * fi) / (ff - fi)
    Call Funcao(qk)
    fk = f
    prod = fk * fi

```

```

    If (prod >= 0) Then
        qi = qk
        fi = fk
    Else
        qf = qk
        ff = fk
    End If
Loop
'Convergência alcançada
'qk é a raiz da função f(qsi) dada na equação (2.5.11) do Volume 3 de Curso de Concreto Armado
,

'Cálculo da taxa mecânica de armadura
If (Math.Abs(soma1) >= Math.Abs(soma2)) Then
    'Uso da equação (2.5.9) do Volume 3
    w = n * fyd * (ani - rc) / soma1
Else
    'Uso da equação (2.5.10) do Volume 3
    w = n * fyd * (0.5 * ani - ami - rc * bc) / soma2
End If
If (w < 0) Then
    w = 0
End If
,

'Cálculo da área de armadura total
100:  aas = w * ac * tcd / fyd
,

'Armadura mínima da NBR-6118 para pilares
,

ani0 = aand / (ac * fcd)
romin = 0.15 * fcd * ani0 / fyd
If (romin < 0.004) Then
    romin = 0.004
End If
asmin = romin * ac
,

'Armadura a ser adotada
ase = aas
If (ase < asmin) Then
    ase = asmin
End If
,

'Convertendo a saída para duas casas decimais
aas = FormatNumber(aas, 2)
asmin = FormatNumber(asmin, 2)
ase = FormatNumber(ase, 2)
,

'Mostrando os resultados
TextBox13.Text = aas
TextBox14.Text = asmin
TextBox15.Text = ase
,

End Sub
Private Sub Funcao(qsi)
    Dim ql, c, esi, tsi As Double
    'Calcula o valor da função f(qsi) dada na equação (2.5.11)
    'do Volume 3 de Curso de Concreto Armado

```

```

'
'qsi=x/h é a profundidade relativa da linha neutra
'rc é a resultante de compressão do concreto adimensional dada na equação (2.4.4)
'bc é a posição da resultante adimensional dada na equação (2.4.5)
'soma1 é o somatório contido no denominador da equação (2.5.9)
'soma2 é o somatório contido no denominador da equação (2.5.10)
'f é o resultado da equação (2.5.11)
'
'O único parâmetro de entrada é qsi, pois as demais variáveis são públicas.
'Os parâmetros rc,bc,soma1,soma2,f são calculados nessa sub-rotina
'
'Constantes para o cálculo das deformações das camadas da armadura
'Observar que o primeiro índice é zero
ql = eu * beta(0) / (eu + 10)
If (qsi <= ql) Then
    'A linha neutra está no domínio 2
    c = 0.01 / (beta(0) - qsi)
ElseIf (qsi <= 1) Then
    'A linha neutra está nos domínios 3,4 e 4a
    c = eu / (1000 * qsi)
Else
    'A linha neutra está no domínio 5
    c = (e0 / 1000) / (qsi - akapa)
End If
'
'Resultante de compressão no concreto
If (qsi < 1 / alamb) Then
    rc = alamb * qsi
    bc = 0.5 * alamb * qsi
Else
    rc = 1
    bc = 0.5
End If
soma1 = 0
soma2 = 0
For i = 0 To nl - 1
    esi = c * (qsi - beta(i))
    Call Tensao(esi)
    tsi = tsl
    soma1 = soma1 + ni(i) * tsi
    soma2 = soma2 + ni(i) * beta(i) * tsi
Next
'Funcao f(qsi)
f = (ami - 0.5 * ani + rc * bc) * soma1 + (ani - rc) * soma2
End Sub
Private Sub Tensao(esl)
    Dim ess, eyd As Double
    '
    'Calcula a tensao no aço
    'es = módulo de elasticidade do aço em kN/cm2
    'esl = deformação de entrada
    'fyd = tensão de escoamento de cálculo em kN/cm2
    'tsl = tensão de saída em kN/cm2
    '
    'Trabalhando com deformação positiva
    ess = Math.Abs(esl)

```



```

eyd = fyd / es
If (ess < eyd) Then
    tsl = es * ess
Else
    tsl = fyd
End If
'Trocando o sinal se necessário
If (esl < 0) Then
    tsl = -tsl
End If
End Sub
Private Sub TextBox10_TextChanged(sender As Object, e As EventArgs) Handles
    TextBox10.TextChanged
        'Essa sub-rotina corresponde à caixa de texto onde é fornecido o número de camadas de armadura
        'Ela faz o dimensionamento dinâmico do DataGridView1
        ,
        nl = Val(TextBox10.Text)
        If (nl > 1) Then
            'Limpando o DataGridView1
            DataGridView1.Rows.Clear()
            'Criando o DataGridView1 com nl linhas
            For i = 0 To nl - 1
                DataGridView1.Rows.Add()
            Next
            'Centralizando o texto nas 2 colunas
            For j = 0 To 1
                DataGridView1.Columns(j).DefaultCellStyle.Alignment =
DataGridViewContentAlignment.MiddleCenter
            Next
            'Preenchendo a primeira coluna
            For i = 0 To nl - 1
                DataGridView1.Item(0, i).Value = i + 1
            Next
        End If
    End Sub
End Class

```

6.5 - Programa em Visual C# 2017

Na fig. 6.5.1 apresenta-se o formulário do programa escrito em Visual C# 2017. O formulário é inteiramente análogo ao do programa em Visual Basic 2017. Os dados podem ser fornecidos usando o ponto ou a vírgula como separador decimal. Os controles são os mesmos do programa em Visual Basic.

O arquivo com a listagem do código-fonte em Visual C# 2017 pode ser obtido em www.editoradunas.com.br/programas.htm. Após a elaboração do formulário, basta clicar com o botão direito do mouse sobre o formulário para Exibir Código e colar o código-fonte.

Fig. 6.5.1 – Formulário do programa em Visual C# 2017

LISTAGEM DO CÓDIGO-FONTE EM VISUAL C# 2017

```
using System;
using System.Globalization;
using System.Windows.Forms;

namespace DIMFCN
{
    public partial class Form1 : Form
    {
        //Declarnado variáveis públicas
        public double[] beta = new double[20];
        public int[] ni = new int[20];
        public double rc, bc, soma1, soma2, f, qsi, tsl;
        public double es, esl, fyd;
        public double delta, ani, ami, fcd;
        public double alfac, alamb, e0, eu, akapa;

        public Form1()
        {
            InitializeComponent();
        }

        private void button1_Click(object sender, EventArgs e)
        {

```

```

double fck, fyk, gamac, gamas, gamaf;
double b, h, dl, ank, amk;
double tcd, aand, amd, ac, w;
double a,aux, tsd0, qi, fi, qf, ff, qk, fk, prod;
double aas, ani0, romin, asmin, ase;
int i, n, nl;
//
//Essa sub-rotina corresponde ao botão CALCULAR
//
//Dimensionamento à Flexo-Compressão Normal
//Seções retangulares com várias camadas de armadura
//O programa está limitado a 20 camadas
//
//
//ENTRADA DE DADOS
//
//Os dados são lidos das caixas de texto do formulário e da grade.
//
//Resistência característica à compressão do concreto em MPa
//Substituindo a vírgula por ponto
textBox1.Text = textBox1.Text.Replace(",", ".");
fck = Convert.ToDouble(textBox1.Text, CultureInfo.InvariantCulture);
//
//Tensão de escoamento característica do aço em MPa
textBox2.Text = textBox2.Text.Replace(",", ".");
fyk = Convert.ToDouble(textBox2.Text, CultureInfo.InvariantCulture);
//
//Módulo de elasticidade do aço em GPa
textBox3.Text = textBox3.Text.Replace(",", ".");
es = Convert.ToDouble(textBox3.Text, CultureInfo.InvariantCulture);
//
//Coeficientes parciais de segurança:
//para o concreto
textBox4.Text = textBox4.Text.Replace(",", ".");
gamac = Convert.ToDouble(textBox4.Text, CultureInfo.InvariantCulture);
//
//para o aço
textBox5.Text = textBox5.Text.Replace(",", ".");
gamas = Convert.ToDouble(textBox5.Text, CultureInfo.InvariantCulture);
//
//para os esforços solicitantes
textBox6.Text = textBox6.Text.Replace(",", ".");
gamaf = Convert.ToDouble(textBox6.Text, CultureInfo.InvariantCulture);
//
//Dados da seção transversal:
//Largura em cm
textBox7.Text = textBox7.Text.Replace(",", ".");
b = Convert.ToDouble(textBox7.Text, CultureInfo.InvariantCulture);
//
//Altura em cm
textBox8.Text = textBox8.Text.Replace(",", ".");
h = Convert.ToDouble(textBox8.Text, CultureInfo.InvariantCulture);
//
//Distância dl
textBox9.Text = textBox9.Text.Replace(",", ".");
dl = Convert.ToDouble(textBox9.Text, CultureInfo.InvariantCulture);
//

```

```

//Número de camadas de armadura:
//A seção deve ter, no mínimo, duas camadas de armadura
//nl>=2

nl = Convert.ToInt16(textBox10.Text, CultureInfo.InvariantCulture);
//
//Número de barras das camadas de armadura:
//As camadas são numeradas de baixo para cima
//Cada camada deve ter, no mínimo, duas barras
for (i = 0; i <= nl-1; i++)
{
    ni[i] = Convert.ToInt16(dataGridView1[1, i].Value, CultureInfo.InvariantCulture);
}
//Esforços solicitantes de serviço:
//Entrar com valores positivos
//Esforço normal (em kN)
textBox11.Text = textBox11.Text.Replace(",", ".");
ank = Convert.ToDouble(textBox11.Text, CultureInfo.InvariantCulture);
//Momento fletor (em kNm)
textBox12.Text = textBox12.Text.Replace(",", ".");
amk = Convert.ToDouble(textBox12.Text, CultureInfo.InvariantCulture);

//
//
//FIM DA ENTRADA DE DADOS
//
//INÍCIO DOS CÁLCULOS
//
//Parâmetros do diagrama retangular
if (fck <= 50)
{
    alamb = 0.8;
    alfac = 0.85;
    eu = 3.5;
    e0 = 2;
}
else
{
    alamb = 0.8 - (fck - 50) / 400;
    alfac = 0.85 * (1 - (fck - 50) / 200);
    a = (90 - fck) / 100;
    eu = 2.6 + 35 * Math.Pow(a, 4);
    aux = fck - 50;
    a = 0.53;
    e0 = 2 + 0.085 * Math.Pow(aux, a);
}
//Parâmetro kapa que define o ponto com deformação igual a eo no domínio 5
akapa = 1 - e0 / eu;
//
//Conversão de unidades: transformando para kN e cm
fck = fck / 10;
fyk = fyk / 10;
es = 100 * es;
amk = 100 * amk;
//
//Resistências de cálculo
fcd = fck / gamac;

```

```

tcd = alfac * fcd;
fyd = fyk / gamas;
//
//Esforços solicitantes de cálculo
aand = gamaf * ank;
amd = gamaf * amk;
//
//Cálculo do número total de barras na seção
n = 0;
for (i = 0; i <= nl-1; i++)
{
    n = n + ni[i];
}
//
//Parâmetro geométrico
delta = dl / h;
//
//Área da seção de concreto
ac = b * h;
//
//Esforços reduzidos
ani = aand / (ac * tcd);
ami = amd / (ac * h * tcd);
//
//Caso particular de compressão centrada
//
if(ami == 0)
{
    esl = e0 / 1000;
    Tensao();
    tsd0 = tsl;
    w = (ani - 1) * fyd / tsd0;
    if (w < 0)
    {
        w = 0;
    }
    goto Finalizar;
}
//
//Flexo-compressão normal
//Montagem do vetor beta
//Ver equação (2.2.5) do Volume 3 de Curso de Concreto Armado
//Aqui a primeira camada tem índice = 0
//A equação foi modificada para compatibilizar
//
for (i = 0; i <= nl-1; i++)
{
    beta[i] = delta + (nl - 1 - i) * (1 - 2 * delta) / (nl - 1);
}
//
//Processo iterativo da bissecante
//
//Determinação do intervalo solução
//
//Valor inicial para a linha neutra adimensional  $q_i = x/h$ 
qi = 0;
//Chamar sub-rotina para calcular o valor da função  $f_i = f(q_i)$ 

```

```

qsi = qi;
Funcao();
fi = f;
//
//Valor final para a linha neutra adimensional qsi=x/h
qf = 1000;
//Chamar sub-rotina para calcular o valor da função ff=f(qf)
qsi = qf;
Funcao();
ff = f;
prod = fi * ff;
//Modificando os extremos do intervalo solução até que prod<=0
while(prod > 0)
{
    qi = qf;
    fi = ff;
    qf = 10 * qf;
    qsi = qf;
    Funcao();
    ff = f;
    prod = fi * ff;
}
//
//O intervalo solução foi definido
//A linha neutra qsi fica entre [qi,qf]
//
//Processo iterativo da bissecante
fk = 1;
while(Math.Abs(fk) > 0.001)
{
    qk = (qi * ff - qf * fi) / (ff - fi);
    qsi = qk;
    Funcao();
    fk = f;
    prod = fk * fi;
    if (prod >= 0)
    {
        qi = qk;
        fi = fk;
    }
    else
    {
        qf = qk;
        ff = fk;
    }
}
//Convergência alcançada
//qk é a raiz da função f(qsi) dada na equação (2.5.11) do Volume 3 de Curso de Concreto
Armado
//
//Cálculo da taxa mecânica de armadura
if(Math.Abs(soma1) >= Math.Abs(soma2))
{
    // Uso da equação (2.5.9)do Volume 3
    w = n * fyd * (ani - rc) / soma1;
}
else

```

```

    {
        // Uso da equação (2.5.10) do Volume 3
        w = n * fyd * (0.5 * ani - ami - rc * bc) / soma2;
    }
    if(w < 0)
    {
        w = 0;
    }
//
Finalizar:
//
//Cálculo da área de armadura total
aas = w * ac * tcd / fyd;
//
//Ardadura mínima da NBR-6118 para pilares
//
ani0 = aand / (ac * fcd);
romin = 0.15 * fcd * ani0 / fyd;
if(romin < 0.004)
{
    romin = 0.004;
}
asmin = romin * ac;
//
//Ardadura a ser adotada
ase = aas;
if(ase < asmin)
{
    ase = asmin;
}
//
//Convertendo a saída para duas casas decimais
decimal saida1 = Decimal.Round(Convert.ToDecimal(aas), 2);
decimal saida2 = Decimal.Round(Convert.ToDecimal(asmin), 2);
decimal saida3 = Decimal.Round(Convert.ToDecimal(ase), 2);
//
//
//Mostrando os resultados
textBox13.Text = Convert.ToString(saida1);
textBox14.Text = Convert.ToString(saida2);
textBox15.Text = Convert.ToString(saida3);
//
}
private void Funcao()
{
    int i, nl;
    double ql, c, tsi;
    //Calcula o valor da função f(qsi)dada na equação (2.5.11) do Volume 3 de Curso de Concreto
Armado
//
//qsi=x/h é a profundidade relativa da linha neutra
//rc é a resultante de compressão do concreto adimensional dada na equação (2.4.4)
//bc é a posição da resultante adimensional dada na equação (2.4.5)
//soma1 é o somatório contido no denominador da equação (2.5.9)
//soma2 é o somatório contido no denominador da equação (2.5.10)
//f é o resultado da equação (2.5.11)
//

```

```

//Todas as variáveis de entrada são públicas
//Os parâmetros rc,bc,soma1,soma2,f são calculados nessa sub-rotina
//
//Número de camadas de armadura
nl = Convert.ToInt16(textBox10.Text, CultureInfo.InvariantCulture);
//Constantes para o cálculo das deformações das camadas da armadura
//Observar que o primeiro índice é zero
ql = eu * beta[0] / (eu + 10);
if(qsi <= ql)
{
    //A linha neutra está no domínio 2
    c = 0.01 / (beta[0] - qsi);
}
else if(qsi <= 1)
{
    //A linha neutra está nos domínios 3,4 e 4a
    c = eu / (1000 * qsi);
}
else
{
    //A linha neutra está no domínio 5
    c = (e0 / 1000) / (qsi - akapa);
}
//
//Resultante de compressão no concreto
if(qsi < 1 / alamb)
{
    rc = alamb * qsi;
    bc = 0.5 * alamb * qsi;
}
else
{
    rc = 1;
    bc = 0.5;
}
soma1 = 0;
soma2 = 0;
for (i = 0; i <= nl-1; i++)
{
    esl = c * (qsi - beta[i]);
    Tensao();
    tsi = tsl;
    soma1 = soma1 + ni[i] * tsi;
    soma2 = soma2 + ni[i] * beta[i] * tsi;
}
//Funcao f(qsi)
f = (ami - 0.5 * ani + rc * bc) * soma1 + (ani - rc) * soma2;
}
//
private void Tensao()
{
    double ess, eyd;
    //
    //Calcula a tensão no aço
    //es = módulo de elasticidade do aço em kN/cm2
    //esl = deformação de entrada
    //fyd = tensão de escoamento de cálculo em kN/cm2

```



```

//tsl = tensão de saída em kN/cm2
//
//Trabalhando com deformação positiva
ess = Math.Abs(esl);
eyd = fyd / es;
if (ess < eyd)
{
    tsl = es * ess;
}
else
{
    tsl = fyd;
}
//Trocando o sinal se necessário
if (esl < 0)
{
    tsl = -tsl;
}
}
//
private void textBox10_TextChanged_1(object sender, EventArgs e)
{
    int i, j, nl;
    //Essa sub-rotina corresponde à caixa de texto onde é fornecido o número de camadas de
armadura
    //Ela faz o dimensionamento dinâmico do DataGridView1
    //
    nl = Convert.ToInt16(textBox10.Text, CultureInfo.InvariantCulture);
    if (nl > 0)
    {
        //Limpando o DataGridView1
        dataGridView1.Rows.Clear();
        //Criando o DataGridView1 com n linhas
        for (i = 0; i <= nl - 1; i++)
        {
            dataGridView1.Rows.Add();
        }
        //Centralizando o texto nas 2 colunas
        for (j = 0; j <= 1; j++)
        {
            dataGridView1.Columns[j].DefaultCellStyle.Alignment =
DataGridViewContentAlignment.MiddleCenter;
        }
        //Preenchendo a primeira coluna
        for (i = 0; i <= nl - 1; i++)
        {
            dataGridView1[0, i].Value = i + 1;
        }
    }
}
//
//
}
}

```

6.6 – Exemplos para testar os programas

Os resultados apresentados a seguir servem para o leitor testar os programas. Os exemplos correspondem às seções das figuras 6.6.1 e 6.6.2. Esses exemplos foram resolvidos manualmente no Capítulo 2 do Volume 3.

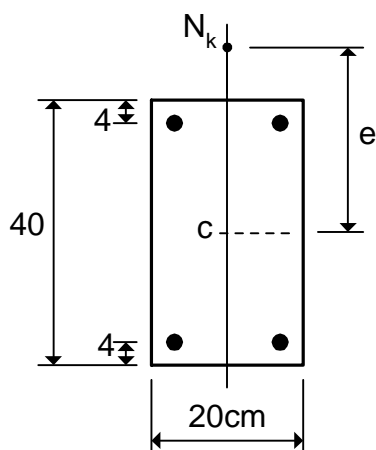


Fig. 6.6.1 - Seção do exemplo 1

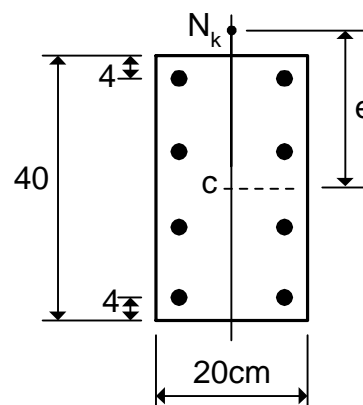


Fig. 6.6.2 - Seção do exemplo 2

São dados do problema:

$$N_k = 410 \text{ kN}; \quad e = 25 \text{ cm}; \quad M_k = N_k e = 102,5 \text{ kNm}$$

Aço CA-50: $f_{yk} = 500 \text{ MPa}$; $E_s = 200 \text{ GPa}$.

$\gamma_c = 1,4$, $\gamma_s = 1,15$ e $\gamma_f = 1,4$.

Na tabela 6.6.1 apresentam-se alguns resultados obtidos com os programas.

Tabela 6.6.1 - Resultados obtidos com os programas

Exemplo 1					
f_{ck} (MPa)	20	30	50	70	90
A_s (cm ²)	15,78*	10,63	8,03	7,43	7,01
Exemplo 2					
f_{ck} (MPa)	20	30	50	70	90
A_s (cm ²)	25,17*	18,53	10,93	9,34	8,50
* Os valores da tabela 2.9.1 do Volume 3 são arredondados.					

Capítulo 7

FLEXO-COMPRESSÃO NORMAL Verificação de Seções Retangulares

7.1 – Apresentação do problema

O objetivo é determinar o momento de ruína de uma seção retangular com várias camadas de armadura, submetida à flexo-compressão normal. A geometria da seção é indicada na fig. 7.1.1. A teoria é apresentada nos Capítulos 2 e 3 do Volume 3.

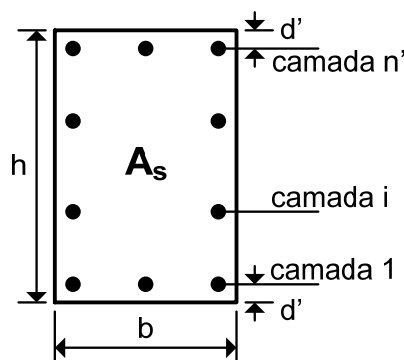


Fig. 7.1.1 - Seção retangular com várias camadas de armadura

Os dados do problema são os seguintes:

- a) propriedades dos materiais: f_{ck} , f_{yk} , E_s
- b) coeficientes parciais de segurança: γ_c , γ_s
- c) dimensões da seção transversal: b , h , d'
- d) área total de aço na seção: A_s
- e) número de camadas de armadura: $n' \geq 2$
- f) número de barras de cada camada: ≥ 2
- g) esforço normal de cálculo: N_d

O valor requerido é o momento fletor de ruína de cálculo M_d .

7.2 – Roteiro para programação

1) Entrada de dados

Ler as propriedades dos materiais: f_{ck} (MPa), f_{yk} (MPa), E_s (GPa)

Ler os coeficientes parciais de segurança: γ_c , γ_s

Ler as dimensões da seção (em cm): b , h , d'

Ler a área de aço existente na seção (em cm²): A_s

Ler o número de camadas de armadura: $n' \geq 2$

Ler o número de barras de cada camada: $n_i(i) \geq 2$

Ler o esforço normal de cálculo (em kN): N_d

Fim da entrada de dados e início dos cálculos.

2) Parâmetros do diagrama retangular para o concreto

- Se $f_{ck} \leq 50$ MPa:

$$\lambda = 0,8$$

$$\alpha_c = 0,85$$

$$\varepsilon_u \left(\frac{o}{oo} \right) = 3,5$$

$$\varepsilon_o \left(\frac{o}{oo} \right) = 2$$

- Se $f_{ck} > 50$ MPa:

$$\lambda = 0,8 - \frac{(f_{ck} - 50)}{400}$$

$$\alpha_c = 0,85 \left[1 - \frac{(f_{ck} - 50)}{200} \right]$$

$$\varepsilon_u \left(\frac{o}{oo} \right) = 2,6 + 35 \left(\frac{90 - f_{ck}}{100} \right)^4$$

$$\varepsilon_o \left(\frac{o}{oo} \right) = 2,0 + 0,085(f_{ck} - 50)^{0,53}$$

3) Parâmetro kapa

$$\kappa = 1 - \varepsilon_o / \varepsilon_u$$

4) Conversão de unidades para kN e cm

$$f_{ck} = f_{ck} / 10$$

$$f_{yk} = f_{yk} / 10$$

$$E_s = 100 E_s$$

5) Resistências de cálculo

$$f_{cd} = f_{ck} / \gamma_c$$

$$\sigma_{cd} = \alpha_c f_{cd}$$

$$f_{yd} = f_{yk} / \gamma_s$$

6) Cálculo do número total de barras na seção transversal

$$n = \sum_{i=1}^{n'} ni(i)$$

7) Parâmetro geométrico e área da seção transversal

$$\delta = d'/h$$

$$A_c = bh$$

8) Taxa mecânica de armadura

$$\omega = \frac{A_s f_{yd}}{A_c \sigma_{cd}}$$

9) Esforço normal reduzido

$$\nu = \frac{N_d}{A_c \sigma_{cd}}$$

10) Esforço normal máximo resistido pela seção

$$\varepsilon_s = \varepsilon_o \text{ (deformação uniforme nas armaduras)}$$

Chamar uma sub-rotina para calcular a tensão σ_{sdo} .

$$\nu_{\max} = 1 + \frac{\omega \sigma_{sdo}}{f_{yd}}$$

Se $\nu > \nu_{\max}$, colocar mensagem informando que a seção não resiste ao esforço normal dado e parar o processamento.

11) Montagem do vetor β

Ver equação (2.2.5) do Capítulo 2 do Volume 3.

$$\beta_i = \delta + \frac{(n' - i)(1 - 2\delta)}{(n' - 1)}, \text{ com } i = 1, \dots, n'$$

12) Determinação do intervalo solução para a linha neutra

Profundidade inicial: $\xi_i = 0$

Chamar sub-rotina para calcular a função $f_i = f(\xi_i)$.

Profundidade final: $\xi_f = 1000$

Chamar sub-rotina para calcular a função $f_f = f(\xi_f)$.

Calcular $prod = f_i f_f$

➤ Fazer enquanto $prod > 0$:

$$\xi_i = \xi_f$$

$$f_i = f_f$$

$$\xi_f = 10\xi_f$$

Chamar sub-rotina para calcular a função $f_f = f(\xi_f)$.

$$prod = f_i f_f$$

⬅ Retornar

A linha neutra está no intervalo $[\xi_i, \xi_f]$.

13) Processo iterativo da bissecante para encontrar a linha neutra

Iterar até a convergência (processo da bissecante):

Fazer: $f_k = 1$

➤ Fazer enquanto $|f_k| > 0.001$:

$$\xi_k = (\xi_i f_f - \xi_f f_i) / (f_f - f_i)$$

Chamar sub-rotina para calcular a função $f_k = f(\xi_k)$.

Calcular $prod = f_k f_i$

• Se $prod \geq 0$

$$\xi_i = \xi_k$$

$$f_i = f_k$$

• Se $prod < 0$

$$\xi_f = \xi_k$$

$$f_f = f_k$$

⬅ Retornar

15) Cálculo do momento fletor reduzido

Emprego da equação (3.2.2) do Capítulo 3 do Volume 3.

$$\mu = 0,5v - r_c \beta_c - \frac{\omega}{nf_{yd}} \text{ soma2}$$

onde $\text{soma2} = \sum_{i=1}^{n'} n_i \beta_i \sigma_{sdi}$ é calculado na Sub-rotina Função.

16) Momento fletor de ruína

$$M_d = \mu A_c h \sigma_{cd}$$

Imprimir ou mostrar o resultado: M_d (em kNcm)

Fim do programa principal.

B) Sub-rotina função

Essa sub-rotina calcula o valor da função $f(\xi)$ dada na equação (3.2.1) do Volume 3. A profundidade relativa da linha neutra, $\xi = x/h$, é um parâmetro de entrada na sub-rotina.

1) Constantes para cálculo das deformações na armadura

$$\xi_1 = \left(\frac{\varepsilon_u}{\varepsilon_u + 10\text{‰}} \right) \beta_1$$

- Se $\xi \leq \xi_1$, a linha neutra está no domínio 2:
 $c = 0,01/(\beta_1 - \xi)$
- Se $\xi_1 < \xi \leq 1$, a linha neutra está nos domínios 3, 4 ou 4a:
 $c = (\varepsilon_u / 1000) / \xi$
- Se $\xi > 1$, a linha neutra está no domínio 5:
 $c = (\varepsilon_o / 1000) / (\xi - \kappa)$

2) Resultante de compressão no concreto

- Se $\xi < 1/\lambda$:
 $r_c = \lambda \xi$
 $\beta_c = 0,5 \lambda \xi$
- Se $\xi \geq 1/\lambda$:
 $r_c = 1$
 $\beta_c = 0,5$

3) Cálculo dos somatórios das equações (3.2.1) e (3.2.2)

$$soma1 = 0$$

$$soma2 = 0$$

► Somar de 1 a n' : $\sum_{i=1}^{n'}$

$$\varepsilon_{si} = c(\xi - \beta_i)$$

Chamar sub-rotina para calcular a tensão σ_{si} .

$$soma1 = soma1 + n(i) \sigma_{si}$$

$$\blacktriangleleft soma2 = soma2 + n(i) \beta_i \sigma_{si}$$

4) Função $f(\xi)$

$$f(\xi) = \nu - r_c - \frac{\omega}{nf_{yd}} \text{ soma1}$$

Fim da sub-rotina.

C) Sub-rotina tensão

Parâmetros de entrada: E_s , ε_s , f_{yd}

Parâmetro de saída: σ_{sd}

1) Trabalhar com o valor absoluto da deformação

$$\varepsilon_{ss} = |\varepsilon_s|$$

2) Deformação de escoamento de cálculo do aço

$$\varepsilon_{yd} = f_{yd} / E_s$$

3) Cálculo da tensão

- Se $\varepsilon_{ss} < \varepsilon_{yd}$:

$$\sigma_{sd} = E_s \varepsilon_{ss}$$

- Se $\varepsilon_{ss} \geq \varepsilon_{yd}$:

$$\sigma_{sd} = f_{yd}$$

4) Acertando o sinal da tensão

Se $\varepsilon_s < 0$, fazer $\sigma_{sd} = -\sigma_{sd}$

Fim da sub-rotina.

7.3 – Programa em Fortran 90

A listagem a seguir apresenta o código do programa em Fortran 90. Os dados são lidos do arquivo DADOS.TXT. Para facilitar a comparação com os programas em Visual Basic e Visual C#, cada dado é lido em uma linha do arquivo. As mensagens de alerta e os resultados são mostrados na tela. Os dados devem ser fornecidos usando somente o ponto como separador decimal. O arquivo com a listagem do código-fonte em Fortran 90 pode ser obtido em www.editoradunas.com.br/programas.htm.

LISTAGEM DO CÓDIGO-FONTE EM FORTRAN 90

```
!  
!  
!Flexo-Compressão Normal: Verificação  
!Seções retangulares com várias camadas de armadura  
!O programa está limitado a 20 camadas  
!  
!  
!PROGRAMA PRINCIPAL  
!  
!Declarando variáveis reais de dupla precisão  
IMPLICIT REAL*8(A-H,O-Z)  
!Variáveis públicas  
COMMON /A1/ ni(20),beta(20)  
COMMON /A2/ n,nl  
COMMON /A3/ delta,ani,ami,fyd,fcd,es  
COMMON /A4/ alfac,alamb,e0,eu,akapa  
!  
!  
!Abertura do arquivo de dados  
OPEN(1,FILE='DADOS.TXT')  
!  
!LEITURA DOS DADOS  
!  
!Os dados são lidos do arquivo DADOS.TXT  
!Cada dado deve estar em uma linha do arquivo  
!Resistência característica à compressão do concreto em MPa  
read(1,*)fck  
!  
!Tensão de escoamento característica do aço em MPa  
read(1,*)fyk  
!  
!Módulo de elasticidade do aço em GPa  
read(1,*)es  
!  
!Coeficientes parciais de segurança:  
!para o concreto  
read(1,*)gamac  
!para o aço  
read(1,*)gamas  
!  
!Dados da seção transversal:  
!Largura em cm  
read(1,*)b  
!Altura em cm  
read(1,*)h  
!Distância d'  
read(1,*)dl  
!  
!Área total de aço na seção  
read(1,*)aas  
!  
!Número de camadas de armadura:
```

```

!A seção deve ter, no mínimo, duas camadas de armadura
!nl>=2
read(1,*)nl
!
!Número de barras das camadas de armadura:
!As camadas são numeradas de baixo para cima
!Cada camada deve ter, no mínimo, duas barras
do i=1,nl
    read(1,*)ni(i)
end do
!
!Esforço normal de cálculo Nd
!Forneça um valor positivo para Nd
read(1,*)aand
!
!
!FIM DA ENTRADA DE DADOS
!
!INÍCIO DOS CÁLCULOS
!
!Parâmetros do diagrama retangular
If(fck<=50) Then
    alamb=0.8
    alfac=0.85
    eu=3.5
    e0=2
Else
    alamb=0.8-(fck-50)/400
    alfac=0.85*(1-(fck-50)/200)
    eu=2.6+35*((90-fck)/100)**4
    e0=2+0.085*((fck-50)**0.53)
End If
!Parâmetro kapa que define o ponto com deformação igual a eo no domínio 5
akapa=1-e0/eu
!
!Conversão de unidades: transformando para kN e cm
fck=fck/10
fyk=fyk/10
es=100*es
!
!Resistências de cálculo
fcd=fck/gamac
tcd=alfac*fcd
fyd=fyk/gamas
!
!Cálculo do número total de barras na seção
n=0
do i=1,nl
    n=n+ni(i)
end do
!
!Parâmetro geométrico
delta=dl/h
!
!Área da seção de concreto
ac=b*h
!

```

```

!Taxa mecânica de armadura
w=aas*fyd/(ac*tcd)
!
!Esforço normal reduzido
ani=aand/(ac*tcd)
!
!Esforço normal máximo que a seção resiste em compressão simples
!
esi=e0/1000
Call Tensao(esi,tsd0)
animax=1+w*tsd0/fyd
!
!Verificação
If (ani>animax)Then
  write(*,10)
  10 format(//,5x,'A secao nao suporta o esforco normal dado.')
  Goto 100
End If
!Montagem do vetor beta
!Ver equação (2.2.5) do Volume 3 de Curso de Concreto Armado
do i=1,nl
  beta(i)=delta+(nl-i)*(1-2*delta)/(nl-1)
end do
!
!Processo iterativo da bissecante
!
!Determinação do intervalo solução
!Valor inicial para a linha neutra adimensional qsi=x/h
qi=0
!Chamar sub-rotina para calcular o valor da função fi=f(qi)
Call Funcao(w,qi,rc,bc,soma1,soma2,fi)
!
!Valor final para a linha neutra adimensional qsi=x/h
qf=1000
!Chamar sub-rotina para calcular o valor da função ff=f(qf)
Call Funcao(w,qf,rc,bc,soma1,soma2,ff)
prod=fi*ff
!Modificando os extremos do intervalo solução até que prod<=0
Do While (prod>0)
  qi=qf
  fi=ff
  qf=10*qf
  Call Funcao(w,qf,rc,bc,soma1,soma2,ff)
  prod=fi*ff
End do
!O intervalo solução foi definido
!A linha neutra qsi fica entre [qi,qf]
!
!Processo iterativo da bissecante
fk = 1
Do While (Abs(fk)>0.001)
  qk=(qi*ff-qf*fi)/(ff-fi)
  Call Funcao(w,qk,rc,bc,soma1,soma2,fk)
  prod=fk*fi
  If (prod>=0) Then
    qi=qk
    fi=fk

```

```

Else
    qf=qk
    ff=fk
End If
End do
!
!Convergência alcançada
!qk é a raiz da função f(qsi) dada na equação (3.2.1) do Volume 3 de Curso de
!Concreto Armado
!
!Cálculo do momento fletor reduzido conforme a equação (3.2.2) do Volume 3
!
ami=0.5*ani-rc*bc-w*soma2/(n*fyd)
!
!Momento fletor dimensional
!
amud=ami*ac*h*tcd
!Passagem para kNm
amud=amud/100
!
!Mostrando o resultado
write(*,20)amud
20 format(//,5x,'Momento fletor de ruina de calculo (kNm)=' ,f7.2)
!
!Encerrando o programa
100 a=0
End
!
!
Subroutine Funcao(w,qsi,rc,bc,soma1,soma2,f)
IMPLICIT REAL*8(A-H,O-Z)
!Variáveis públicas
COMMON /A1/ ni(20),beta(20)
COMMON /A2/ n,nl
COMMON /A3/ delta,ani,ami,fyd,fcd,es
COMMON /A4/ alfac,alamb,e0,eu,akapa
!Calcula o valor da função f(qsi) dada na equação (3.2.1) do Volume 3 de Curso de
!Concreto Armado
!
!w é a taxa mecânica de armadura
!qsi=x/h é a profundidade relativa da linha neutra
!rc é a resultante de compressão do concreto adimensional dada na equação (2.4.4)
!bc é a posição da resultante adimensional dada na equação (2.4.5)
!soma1 é o somatório que aparece na equação (3.2.1)
!soma2 é o somatatório que aparece na equação (3.2.2)
!f é o resultado da equação (3.2.1)
!
!Os parâmetro de entrada são w e qsi, pois as demais variáveis são públicas.
!Os parâmetros rc,bc,soma1,soma2,f são calculados nessa sub-rotina
!
!
!Constantes para o cálculo das deformações das camadas da armadura
ql= eu*beta(1)/(eu+10)
If (qsi<=ql)Then
    !A linha neutra está no domínio 2
    c=0.01/(beta(1)-qsi)
ElseIf (qsi<=1)Then

```

```

!A linha neutra está nos domínios 3,4 e 4a
c=eu/(1000*qli)
Else
!A linha neutra está no domínio 5
c=(e0/1000)/(qli-akapa)
End If
!Resultante de compressão no concreto
If (qli<1/alamb)Then
rc=alamb*qli
bc=0.5*alamb*qli
Else
rc=1
bc=0.5
End If
!
soma1=0
soma2=0
do i=1,nl
esi=c*(qli-beta(i))
Call Tensao(esi,tsi)
soma1=soma1+ni(i)*tsi
soma2=soma2+ni(i)*beta(i)*tsi
end do
!Funcao f(qli)
f=ani-rc-w*soma1/(n*fyd)
Return
End
!
!
Subroutine Tensao(esl, tsl)
IMPLICIT REAL*8(A-H,O-Z)
!Variáveis públicas
COMMON /A3/ delta,ani,ami,fyd,fcd,es
!
!Calcula a tensão no aço
!es = módulo de elasticidade do aço em kN/cm2
!esl = deformação de entrada
!fyd = tensão de escoamento de cálculo em kN/cm2
!tsl = tensão de saída em kN/cm2
!
!Trabalhando com deformação positiva
ess=Abs(esl)
eyd=fyd/es
If (ess<eyd)Then
tsl= es*ess
Else
tsl=fyd
End If
!Trocando o sinal se necessário
If (esl<0) Then
tsl=-tsl
End If
Return
End

```

7.4 – Programa em Visual Basic 2017

Na fig. 7.4.1 apresenta-se o formulário do programa escrito em Visual Basic 2017. Os dados podem ser fornecidos usando o ponto ou a vírgula como separador decimal. Os dados das camadas de armadura são fornecidos em uma grade, a qual é exibida assim que o usuário digitar o número de camadas.

The screenshot shows a Windows application window titled "Flexo-Compressão Normal - Verificação de Seções Retangulares". On the left, there is a diagram of a rectangular cross-section of a reinforced concrete beam. The diagram labels the width as b , the total height as h , the effective depth as d , and the distance from the top reinforcement to the top fiber as d' . It also shows reinforcement layers labeled "camada 1", "camada i", and "camada n'". The area of steel is denoted as A_s . Below the diagram is a table with two columns: "Camada" and "Número de barras". The main area of the form contains several input fields grouped into sections: "Materiais" (with f_{ck} (MPa), f_{yk} (MPa), and E_s (GPa)), "Coeficientes parciais" (with γ_{mac} and γ_{mas}), "Seção transversal" (with b (cm), h (cm), d' (cm), A_s (cm²), and "Número de camadas: n' "), "Esforço normal de cálculo" (with N_d (kN)), and "Momento de ruína de cálculo" (with M_d (kNm)). A "CALCULAR" button is located at the bottom right.

Fig. 7.4.1 – Formulário do programa em Visual Basic 2017

Para criar o projeto, procede-se como descrito na seção 1.4 do capítulo 1. Utilizando a Caixa de Ferramentas, são introduzidos os controles no Form1. A propriedade Text do Form1 é "Flexo-Compressão Normal – Verificação de Seções Retangulares", como aparece no topo do formulário.

O arquivo com a listagem do código-fonte em Visual Basic 2017 pode ser obtido em www.editoradunas.com.br/programas.htm. Após a elaboração do formulário, basta clicar com o botão direito do mouse sobre o formulário para Exibir Código e colar o código-fonte.

O formulário contém os seguintes controles:

Tipo	Nome	Descrição
PictureBox	PictureBox1	Imagem da seção
Label	Label1	f_{ck} (MPa)=
TextBox	TextBox1	Valor de f_{ck}
Label	Label2	f_{yk} (MPa)=
TextBox	TextBox2	Valor de f_{yk}
Label	Label3	E_s (GPa)=
TextBox	TextBox3	Valor de E_s
Label	Label4	γ_{mac} =
TextBox	TextBox4	Valor de γ_{mac}
Label	Label5	γ_{mas} =
TextBox	TextBox5	Valor de γ_{mas}
Label	Label6	b (cm)=
TextBox	TextBox6	Valor de b
Label	Label7	h (cm)=

TextBox	TextBox7	Valor de h
Label	Label8	d' (cm)=
TextBox	TextBox8	Valor de d'
Label	Label9	Área de aço: As (cm2)=
TextBox	TextBox9	Valor de As
Label	Label10	Número de camadas: n'=
TextBox	TextBox10	Valor de n'
Label	Label11	Nd (kN)=
TextBox	TextBox11	Valor de Nd
Label	Label12	Md (kNm)=
TextBox	TextBox12	Valor de Md
DataGridView	DataGridView1	Dados das camadas de aço
Button	Button1	CALCULAR
GroupBox	GroupBox1	Materiais
GroupBox	GroupBox2	Coeficientes parciais
GroupBox	GroupBox3	Seção transversal
GroupBox	GroupBox4	Esforço normal de cálculo
GroupBox	GroupBox5	Momento de ruína de cálculo

Para o controle DataGridView1, Habilitar Edição. Para a coluna 1, definir a propriedade ReadOnly como True.

LISTAGEM DO CÓDIGO-FONTE EM VISUAL BASIC 2017

Public Class Form1

Public rc, bc, soma1, soma2, f, tsl As Double

Public beta(20) As Double

Public n, nl, ni(20) As Integer

Public delta, ani, ami, fyd, fcd, es As Double

Public alfac, alamb, e0, eu, akapa As Double

Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click

Dim fck, fyk, gamac, gamas As Double

Dim b, h, dl, aas, aand As Double

Dim tcd, ac, w, esi, tsd0, animax As Double

Dim qi, fi, qf, ff, prod, qk, fk As Double

Dim amud As Double

,

'Essa sub-rotina corresponde ao botão CALCULAR

,

'Flexo-Compressão Normal: Verificação

'Seções retangulares com várias camadas de armadura

'O programa está limitado a 20 camadas

,

,

'ENTRADA DE DADOS

,

'Os dados são lidos das caixas de texto do formulário e da grade.

,

'Resistência característica à compressão do concreto em MPa

'Substituindo a vírgula por ponto


```

TextBox1.Text = Replace(TextBox1.Text, ",", ".")
fck = Val(TextBox1.Text)
'
'Tensão de escoamento característica do aço em MPa
TextBox2.Text = Replace(TextBox2.Text, ",", ".")
fyk = Val(TextBox2.Text)
'
'Módulo de elasticidade do aço em GPa
TextBox3.Text = Replace(TextBox3.Text, ",", ".")
es = Val(TextBox3.Text)
'
'Coeficientes parciais de segurança:
'para o concreto
TextBox4.Text = Replace(TextBox4.Text, ",", ".")
gamac = Val(TextBox4.Text)
'para o aço
TextBox5.Text = Replace(TextBox5.Text, ",", ".")
gamas = Val(TextBox5.Text)
'
'Dados da seção transversal:
'Largura em cm
TextBox6.Text = Replace(TextBox6.Text, ",", ".")
b = Val(TextBox6.Text)
'Altura em cm
TextBox7.Text = Replace(TextBox7.Text, ",", ".")
h = Val(TextBox7.Text)
'Distância d'
TextBox8.Text = Replace(TextBox8.Text, ",", ".")
dl = Val(TextBox8.Text)
'
'Área total de aço na seção
TextBox9.Text = Replace(TextBox9.Text, ",", ".")
aas = Val(TextBox9.Text)
'
'Número de camadas de armadura:
'A seção deve ter, no mínimo, duas camadas de armadura
'nl>=2
nl = Val(TextBox10.Text)
'
'Número de camadas de armadura:
'A seção deve ter, no mínimo, duas camadas de armadura
'nl>=2
nl = Val(TextBox10.Text)
'
'Número de barras das camadas de armadura:
'As camadas são numeradas de baixo para cima
'Cada camada deve ter, no mínimo, duas barras
For i = 0 To nl - 1
    'Lendo os dados no DataGridView
    ni(i) = DataGridView1.Item(1, i).Value
Next
'
'Esforço normal de cálculo Nd
'Forneça um valor positivo para Nd
TextBox11.Text = Replace(TextBox11.Text, ",", ".")
aand = Val(TextBox11.Text)
'

```

```

',
'FIM DA ENTRADA DE DADOS
',
'INÍCIO DOS CÁLCULOS
',
'Parâmetros do diagrama retangular
If (fck <= 50) Then
    alamb = 0.8
    alfac = 0.85
    eu = 3.5
    e0 = 2
Else
    alamb = 0.8 - (fck - 50) / 400
    alfac = 0.85 * (1 - (fck - 50) / 200)
    eu = 2.6 + 35 * ((90 - fck) / 100) ^ 4
    e0 = 2 + 0.085 * ((fck - 50) ^ 0.53)
End If
'Parâmetro kapa que define o ponto com deformação igual a eo no domínio 5
akapa = 1 - e0 / eu
',
'Conversão de unidades: transformando para kN e cm
fck = fck / 10
fyk = fyk / 10
es = 100 * es
',
'Resistências de cálculo
fcd = fck / gamac
tcd = alfac * fcd
fyd = fyk / gamas
',
'Cálculo do número total de barras na seção
n = 0
For i = 0 To nl - 1
    n = n + ni(i)
Next
',
'Parâmetro geométrico
delta = dl / h
',
'Área da seção de concreto
ac = b * h
',
'Taxa mecânica de armadura
w = aas * fyd / (ac * tcd)
',
'Esforço normal reduzido
ani = aand / (ac * tcd)
',
'Esforço normal máximo que a seção resiste em compressão simples
',
esi = e0 / 1000
Call Tensao(esi)
tsd0 = tsl
animax = 1 + w * tsd0 / fyd
',
'Verificação
If (ani > animax) Then

```

```

    MsgBox("A seção não suporta o esforço normal dado.", vbOKOnly + vbExclamation)
    Exit Sub
End If
'Montagem do vetor beta
'Ver equação (2.2.5) do Volume 3 de Curso de Concreto Armado
'Aqui a primeira camada tem índice zero
'A equação foi modificada para compatibilizar
For i = 0 To nl - 1
    beta(i) = delta + (nl - 1 - i) * (1 - 2 * delta) / (nl - 1)
Next
'
'Processo iterativo da bissecante
'
'Determinação do intervalo solução
'Valor inicial para a linha neutra adimensional qsi=x/h
qi = 0
'Chamar sub-rotina para calcular o valor da função fi=f(qi)
Call Funcao(w, qi)
fi = f
'
'Valor final para a linha neutra adimensional qsi=x/h
qf = 1000
'Chamar sub-rotina para calcular o valor da função ff=f(qf)
Call Funcao(w, qf)
ff = f
prod = fi * ff
'Modificando os extremos do intervalo solução até que prod<=0
Do While (prod > 0)
    qi = qf
    fi = ff
    qf = 10 * qi
    Call Funcao(w, qf)
    ff = f
    prod = fi * ff
Loop
'O intervalo solução foi definido
'A linha neutra qsi fica entre [qi,qf]
'
'Processo iterativo da bissecante
fk = 1
Do While (Math.Abs(fk) > 0.001)
    qk = (qi * ff - qf * fi) / (ff - fi)
    Call Funcao(w, qk)
    fk = f
    prod = fk * fi
    If (prod >= 0) Then
        qi = qk
        fi = fk
    Else
        qf = qk
        ff = fk
    End If
Loop
'
'Convergência alcançada
'qk é a raiz da função f(qsi) dada na equação (3.2.1) do Volume 3 de Curso de Concreto Armado
'
```

```

'Cálculo do momento fletor reduzido conforme a equação (3.2.2) do Volume 3
,
ami = 0.5 * ani - rc * bc - w * soma2 / (n * fyd)
,
'Momento fletor dimensional
,
amud = ami * ac * h * tcd
'Passagem para kNm
amud = amud / 100
,
'Convertendo a saída para duas casas decimais
amud = FormatNumber(amud, 2)
,
'Mostrando o resultado
TextBox12.Text = amud
,

End Sub
,
Private Sub Funcao(w, qsi)
    Dim ql, c, esi, tsi As Double
    'Calcula o valor da função f(qsi) dada na equação (3.2.1) do Volume 3 de Curso de Concreto
    Armado
    ,
    'w é a taxa mecânica de armadura
    'qsi=x/h é a profundidade relativa da linha neutra
    'rc é a resultante de compressão do concreto
    'adimensional dada na equação (2.4.4)
    'bc é a posição da resultante adimensional dada na equação (2.4.5)
    'soma1 é o somatório que aparece na equação (3.2.1)
    'soma2 é o somatório que aparece na equação (3.2.2)
    'f é o resultado da equação (3.2.1)
    ,
    'Os parâmetro de entrada são w e qsi, pois as demais variáveis são públicas.
    'Os parâmetros rc,bc,soma1,soma2,f são calculados nessa sub-rotina
    ,
    ,
    'Constantes para o cálculo das deformações das camadas da armadura
    'Aqui a primeira camada tem índice zero
    ql = eu * beta(0) / (eu + 10)
    If (qsi <= ql) Then
        'A linha neutra está no domínio 2
        c = 0.01 / (beta(0) - qsi)
    ElseIf (qsi <= 1) Then
        'A linha neutra está nos domínios 3,4 e 4a
        c = eu / (1000 * qsi)
    Else
        'A linha neutra está no domínio 5
        c = (e0 / 1000) / (qsi - akapa)
    End If
    'Resultante de compressão no concreto
    If (qsi < 1 / alamb) Then
        rc = alamb * qsi
        bc = 0.5 * alamb * qsi
    Else
        rc = 1
        bc = 0.5
    End If

```

```

End If
'
soma1 = 0
soma2 = 0
For i = 0 To nl - 1
    esi = c * (qsi - beta(i))
    Call Tensao(esi)
    tsi = tsl
    soma1 = soma1 + ni(i) * tsi
    soma2 = soma2 + ni(i) * beta(i) * tsi
Next
'Funcao f(qsi)
f = ani - rc - w * soma1 / (n * fyd)
End Sub
Private Sub Tensao(esl)
    Dim ess, eyd As Double
    '
    'Calcula a tensão no aço
    'es = módulo de elasticidade do aço em kN/cm2
    'esl = deformação de entrada
    'fyd = tensão de escoamento de cálculo
    'em kN/cm2
    'tsl = tensão de saída em kN/cm2
    '
    'Trabalhando com deformação positiva
    ess = Math.Abs(esl)
    eyd = fyd / es
    If (ess < eyd) Then
        tsl = es * ess
    Else
        tsl = fyd
    End If
    'Trocando o sinal se necessário
    If (esl < 0) Then
        tsl = -tsl
    End If
End Sub
Private Sub TextBox10_TextChanged(sender As Object, e As EventArgs) Handles
    TextBox10.TextChanged
    'Essa sub-rotina corresponde à caixa de texto onde é fornecido o número de camadas de armadura
    'Ela faz o dimensionamento dinâmico do DataGridView1
    '
    nl = Val(TextBox10.Text)
    If (nl > 1) Then
        'Limpando o DataGridView1
        DataGridView1.Rows.Clear()
        'Criando o DataGridView1 com nl linhas
        For i = 0 To nl - 1
            DataGridView1.Rows.Add()
        Next
        'Centralizando o texto nas 2 colunas
        For j = 0 To 1
            DataGridView1.Columns(j).DefaultCellStyle.Alignment =
DataGridViewContentAlignment.MiddleCenter
        Next
        'Preenchendo a primeira coluna
        For i = 0 To nl - 1

```

```

        DataGridView1.Item(0, i).Value = i + 1
    Next
End If
End Sub

```

```
End Class
```

7.5 - Programa em Visual C# 2017

Na fig. 7.5.1 apresenta-se o formulário do programa escrito em Visual C# 2017. O formulário é inteiramente análogo ao do programa em Visual Basic 2017. Os dados podem ser fornecidos usando o ponto ou a vírgula como separador decimal. Os controles são os mesmos do programa em Visual Basic.

O arquivo com a listagem do código-fonte em Visual C# 2017 pode ser obtido em www.editoradunas.com.br/programas.htm. Após a elaboração do formulário, basta clicar com o botão direito do mouse sobre o formulário para Exibir Código e colar o código-fonte.

The screenshot shows a Windows application window titled "Flexo-Compressão Normal - Verificação de Seções Retangulares". On the left, there is a diagram of a rectangular cross-section with width b , height h , effective depth d' , and reinforcement area A_s . The diagram also labels "camada n'", "camada i", and "camada 1". Below the diagram is a table with two columns: "Camada" and "Número de barras". The main area of the form contains several input fields grouped into sections: "Materiais" (fck (MPa)=, fyk (MPa)=, Es (GPa)=), "Coeficientes parciais" (Gamac =, Gamas =), "Seção transversal" (b (cm)=, h (cm)=, d' (cm)=, Área de aço: As (cm²)=, Número de camadas: n'=), "Esforço normal de cálculo" (Nd (kN)=), and "Momento de ruína de cálculo" (Md (kNm)=). A "CALCULAR" button is located at the bottom right.

Fig. 7.5.1 – Formulário do programa em Visual C# 2017

LISTAGEM DO CÓDIGO-FONTE EM VISUAL C# 2017

```

using System;
using System.Globalization;
using System.Windows.Forms;

namespace VERFCN
{
    public partial class Form1 : Form
    {

```

```

//Declarando variáveis públicas
public double[] beta = new double[20];
public int[] ni = new int[20];
public double w,rc, bc, soma1, soma2, f, qsi, tsl;
public int n;
public double es, esl, fyd;
public double delta, ani, ami, fcd;
public double alfac, alamb, e0, eu, akapa;

public Form1()
{
    InitializeComponent();
}

private void button1_Click(object sender, EventArgs e)
{
    double fck, fyk, gamac, gamas;
    double b, h, dl, aas,aand;
    double tcd, ac, tsd0, animax;
    double qi, fi, qf, ff, prod, qk, fk;
    double a, aux, amud;
    int i, nl;
    //
    //Essa sub-rotina corresponde ao botão CALCULAR
    //
    //Flexo-Compressão Normal: Verificação
    //Seções retangulares com várias camadas de armadura
    //O programa está limitado a 20 camadas
    //
    //
    //ENTRADA DE DADOS
    //
    //Os dados são lidos das caixas de texto do formulário e da grade.
    //
    //Resistência característica à compressão do concreto em MPa
    //Substituindo a vírgula por ponto
    textBox1.Text = textBox1.Text.Replace(",", ".");
    fck = Convert.ToDouble(textBox1.Text, CultureInfo.InvariantCulture);
    //
    //Tensão de escoamento característica do aço em MPa
    textBox2.Text = textBox2.Text.Replace(",", ".");
    fyk = Convert.ToDouble(textBox2.Text, CultureInfo.InvariantCulture);
    //
    //Módulo de elasticidade do aço em GPa
    textBox3.Text = textBox3.Text.Replace(",", ".");
    es = Convert.ToDouble(textBox3.Text, CultureInfo.InvariantCulture);
    //
    //Coeficientes parciais de segurança:
    //para o concreto
    textBox4.Text = textBox4.Text.Replace(",", ".");
    gamac = Convert.ToDouble(textBox4.Text, CultureInfo.InvariantCulture);
    //
    //para o aço
    textBox5.Text = textBox5.Text.Replace(",", ".");
    gamas = Convert.ToDouble(textBox5.Text, CultureInfo.InvariantCulture);
    //
    //Dados da seção transversal:

```

```

//Largura em cm
textBox6.Text = textBox6.Text.Replace(",", ".");
b = Convert.ToDouble(textBox6.Text, CultureInfo.InvariantCulture);
//
//Altura em cm
textBox7.Text = textBox7.Text.Replace(",", ".");
h = Convert.ToDouble(textBox7.Text, CultureInfo.InvariantCulture);
//
//Distância dl
textBox8.Text = textBox8.Text.Replace(",", ".");
dl = Convert.ToDouble(textBox8.Text, CultureInfo.InvariantCulture);
//
//Área total de aço na seção
textBox9.Text = textBox9.Text.Replace(",", ".");
aas = Convert.ToDouble(textBox9.Text, CultureInfo.InvariantCulture);
//Número de camadas de armadura:
//A seção deve ter, no mínimo, duas camadas de armadura
//nl>=2
nl = Convert.ToInt16(textBox10.Text, CultureInfo.InvariantCulture);
//
//Número de barras das camadas de armadura:
//As camadas são numeradas de baixo para cima
//Cada camada deve ter, no mínimo, duas barras
for (i = 0; i <= nl - 1; i++)
{
    ni[i] = Convert.ToInt16(dataGridView1[1, i].Value, CultureInfo.InvariantCulture);
}
//
//Esforço normal de cálculo Nd
//Forneça um valor positivo para Nd
//
textBox11.Text = textBox11.Text.Replace(",", ".");
aand = Convert.ToDouble(textBox11.Text, CultureInfo.InvariantCulture);
//
//
//FIM DA ENTRADA DE DADOS
//
//INÍCIO DOS CÁLCULOS
//
//Parâmetros do diagrama retangular
if (fck <= 50)
{
    alamb = 0.8;
    alfac = 0.85;
    eu = 3.5;
    e0 = 2;
}
else
{
    alamb = 0.8 - (fck - 50) / 400;
    alfac = 0.85 * (1 - (fck - 50) / 200);
    a = (90 - fck) / 100;
    eu = 2.6 + 35 * Math.Pow(a, 4);
    aux = fck - 50;
    a = 0.53;
    e0 = 2 + 0.085 * Math.Pow(aux, a);
}

```



```

//Parâmetro kapa que define o ponto com deformação igual a eo no
//domínio 5
akapa = 1 - e0 / eu;
//
//Conversão de unidades: transformando para kN e cm
fck = fck / 10;
fyk = fyk / 10;
es = 100 * es;
//
//Resistências de cálculo
fcd = fck / gamac;
tcd = alfac * fcd;
fyd = fyk / gamas;
//
//Cálculo do número total de barras na seção
n = 0;
for (i = 0; i <= nl-1; i++)
{
    n = n + ni[i];
}
//
//Parâmetro geométrico
delta = dl / h;
//
//Área da seção de concreto
ac = b * h;
//
//Taxa mecânica de armadura
w = aas * fyd / (ac * tcd);
//
//Esforço normal reduzido
ani = aand / (ac * tcd);
//
//Esforço normal máximo que a seção resiste em compressão simples
//
esl = e0 / 1000;
Tensao();
tsd0 = tsl;
animax = 1 + w * tsd0 / fyd;
//
//Verificação
if(ani > animax)
{
    MessageBox.Show("A seção não suporta o esforço normal dado.");
    return;
}
//Montagem do vetor beta
//Ver equação (2.2.5) do Volume 3 de Curso de Concreto Armado
//Aqui a primeira camada tem índice = 0
//A equação foi modificada para compatibilizar
//
for (i = 0; i <= nl-1; i++)
{
    beta[i] = delta + (nl - 1 - i) * (1 - 2 * delta) / (nl - 1);
}
//
//Processo iterativo da bissecante

```

```

//
//Determinação do intervalo solução
//
//Valor inicial para a linha neutra adimensional  $q_{si}=x/h$ 
 $q_i = 0$ ;
//Chamar sub-rotina para calcular o valor da função  $f_i=f(q_i)$ 
 $q_{si} = q_i$ ;
Funcao();
 $f_i = f$ ;
//
//Valor final para a linha neutra adimensional  $q_{si}=x/h$ 
 $q_f = 1000$ ;
//Chamar sub-rotina para calcular o valor da função  $f_f=f(q_f)$ 
 $q_{si} = q_f$ ;
Funcao();
 $f_f = f$ ;
 $prod = f_i * f_f$ ;
//Modificando os extremos do intervalo solução até que  $prod \leq 0$ 
while ( $prod > 0$ )
{
     $q_i = q_f$ ;
     $f_i = f_f$ ;
     $q_f = 10 * q_f$ ;
     $q_{si} = q_f$ ;
    Funcao();
     $f_f = f$ ;
     $prod = f_i * f_f$ ;
}
//O intervalo solução foi definido
//A linha neutra  $q_{si}$  fica entre  $[q_i, q_f]$ 
//
//Processo iterativo da bissecante
 $fk = 1$ ;
while ( $Math.Abs(fk) > 0.001$ )
{
     $q_k = (q_i * f_f - q_f * f_i) / (f_f - f_i)$ ;
     $q_{si} = q_k$ ;
    Funcao();
     $fk = f$ ;
     $prod = fk * f_i$ ;
    if ( $prod \geq 0$ )
    {
         $q_i = q_k$ ;
         $f_i = fk$ ;
    }
    else
    {
         $q_f = q_k$ ;
         $f_f = fk$ ;
    }
}
//Convergência alcançada
// $q_k$  é a raiz da função  $f(q_{si})$  dada na equação (3.2.1) do Volume 3 de Curso de Concreto
Armado
//
//Cálculo do momento fletor reduzido conforme a equação (3.2.2) do Volume 3
//

```

```

ami = 0.5 * ani - rc * bc - w * soma2 / (n * fyd);
//
//Momento fletor dimensional
//
amud = ami * ac * h * tcd;
//Passagem para kNm
amud = amud / 100;
//
//Convertendo a saída para duas casas decimais
decimal saida = Decimal.Round(Convert.ToDecimal(amud), 2);
//
//Mostrando o resultado
textBox12.Text = Convert.ToString(saida);
//
}
private void Funcao()
{
    int i, nl;
    double ql, c, tsi;
    //Calcula o valor da funcao f(qsi)dada na equacao (3.2.1) do Volume 3
    //de Curso de Concreto Armado
    //
    //w é a taxa mecânica de armadura
    //qsi=x/h é a profundidade relativa da linha neutra
    //rc é a resultante de compressão do concreto adimensional dada na equação (2.4.4)
    //bc é a posição da resultante adimensional dada na equação (2.4.5)
    //soma1 é o somatório contido no denominador da equação (3.2.1)
    //soma2 é o somatório contido no denominador da equação (3.2.2)
    //f é o resultado da equação (3.2.1)
    //
    //Todas as variáveis de entrada são públicas
    //Os parâmetros rc,bc,soma1,soma2,f são calculados nessa sub-rotina
    //
    //Número de camadas de armadura
    nl = Convert.ToInt16(textBox10.Text, CultureInfo.InvariantCulture);
    //Constantes para o cálculo das deformações das camadas da armadura
    //Observar que o primeiro índice é zero
    ql = eu * beta[0] / (eu + 10);
    if (qsi <= ql)
    {
        //A linha neutra está no domínio 2
        c = 0.01 / (beta[0] - qsi);
    }
    else if (qsi <= 1)
    {
        //A linha neutra está nos domínios 3,4 e 4a
        c = eu / (1000 * qsi);
    }
    else
    {
        //A linha neutra está no domínio 5
        c = (e0 / 1000) / (qsi - akapa);
    }
    //
    //Resultante de compressão no concreto
    if (qsi < 1 / alamb)
    {

```

```

        rc = alamb * qsi;
        bc = 0.5 * alamb * qsi;
    }
    else
    {
        rc = 1;
        bc = 0.5;
    }
    soma1 = 0;
    soma2 = 0;
    for (i = 0; i <= nl-1; i++)
    {
        esl = c * (qsi - beta[i]);
        Tensao();
        tsi = tsl;
        soma1 = soma1 + ni[i] * tsi;
        soma2 = soma2 + ni[i] * beta[i] * tsi;
    }
    //Funcao f(qsi)
    f = ani - rc - w * soma1 / (n * fyd);
}

private void Tensao()
{
    double ess, eyd;
    //
    //Calcula a tensão no aço
    //es = módulo de elasticidade do aço em kN/cm2
    //esl = deformação de entrada
    //fyd = tensão de escoamento de cálculo em kN/cm2
    //tsl = tensão de saída em kN/cm2
    //
    //Trabalhando com deformação positiva
    ess = Math.Abs(esl);
    eyd = fyd / es;
    if (ess < eyd)
    {
        tsl = es * ess;
    }
    else
    {
        tsl = fyd;
    }
    //Trocando o sinal se necessário
    if (esl < 0)
    {
        tsl = -tsl;
    }
}

private void textBox10_TextChanged(object sender, EventArgs e)
{
    int i, j, nl;
    //Essa sub-rotina corresponde à caixa de texto onde é fornecido o número de camadas de
    armadura
    //Ela faz o dimensionamento dinâmico do DataGridView1
    //

```

```

nl = Convert.ToInt16(textBox10.Text, CultureInfo.InvariantCulture);
if (nl > 0)
{
    //Limpando o DataGridView1
    dataGridView1.Rows.Clear();
    //Criando o DataGridView1 com n linhas
    for (i = 0; i <= nl - 1; i++)
    {
        dataGridView1.Rows.Add();
    }
    //Centralizando o texto nas 2 colunas
    for (j = 0; j <= 1; j++)
    {
        dataGridView1.Columns[j].DefaultCellStyle.Alignment
DataGridViewContentAlignment.MiddleCenter;
    }
    //Preenchendo a primeira coluna
    for (i = 0; i <= nl - 1; i++)
    {
        dataGridView1[0, i].Value = i + 1;
    }
    }
}
}

```

7.6 – Exemplos para testar os programas

Os resultados apresentados a seguir servem para o leitor testar os programas. Os exemplos correspondem às seções das figuras 7.6.1 e 7.6.2.

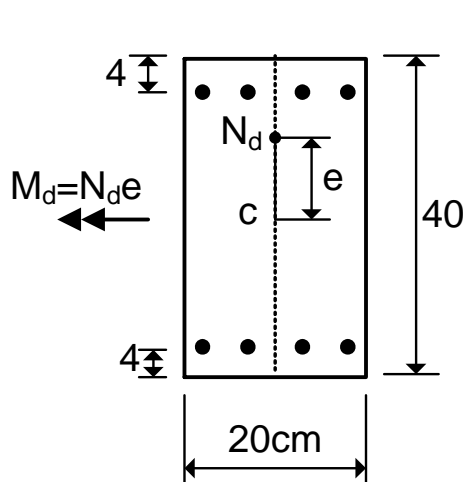


Fig. 7.6.1 - Seção do exemplo 1

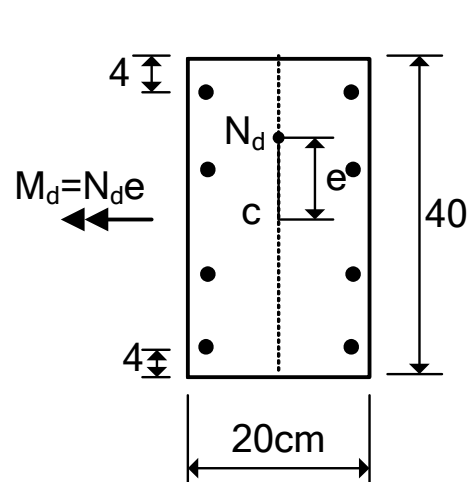


Fig. 7.6.2 - Seção do exemplo 2

São dados do problema:

$$\begin{aligned}
 N_d &= 560 \text{ kN}; & A_s &= 16 \text{ cm}^2 \\
 \text{Aço CA-50: } f_{yk} &= 500 \text{ MPa}; & E_s &= 200 \text{ GPa.} \\
 \gamma_c &= 1,4, & \gamma_s &= 1,15.
 \end{aligned}$$

Na tabela 7.6.1 apresentam-se alguns resultados obtidos com os programas.

Tabela 7.6.1 - Momentos fletores de ruína de cálculo

Exemplo 1					
f_{ck} (MPa)	20	30	50	70	90
M_d (kNm)	146,09	180,14	197,33	199,83	202,64
Exemplo 2					
f_{ck} (MPa)	20	30	50	70	90
M_d (kNm)	107,04	133,20	167,08	176,20	180,98

Capítulo 8

DIMENSIONAMENTO À TORÇÃO

8.1 – Apresentação do problema

O objetivo é realizar o dimensionamento de uma seção retangular submetida à torção combinada com flexão. A geometria da seção é indicada na fig. 8.1.1. A teoria é apresentada no Capítulo 4 do Volume 1 (para a flexão simples), no Capítulo 7 do Volume 1 (para o esforço cortante) e no Capítulo 1 do Volume 4 (para a torção).

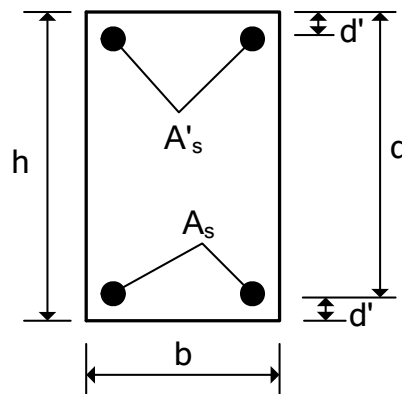


Fig. 8.1.1 - Seção transversal

Os dados do problema são os seguintes:

- a) propriedades dos materiais: f_{ck} , f_{yk} , E_s
- b) coeficientes parciais de segurança: γ_c , γ_s , γ_f
- c) coeficiente de redistribuição de momentos: $0,75 \leq \beta \leq 1$
- d) dimensões da seção transversal: b , h , d'
- e) momento fletor de serviço: M_k
- f) esforço cortante de serviço: V_k
- g) momento torçor de serviço: T_k

Os valores requeridos são as áreas dos estribos e das armaduras longitudinais, considerando os três esforços solicitantes.

8.2 – Roteiro para programação

A) Programa principal

1) Entrada de dados

Ler as propriedades dos materiais: f_{ck} (MPa), f_{yk} (MPa), E_s (GPa)

Ler os coeficientes parciais de segurança: γ_c , γ_s , γ_f

Ler o coeficiente de redistribuição de momentos: β

Ler as dimensões da seção (em cm): b , h , d'

Ler o momento fletor de serviço (em kNm): M_k

Ler o esforço cortante de serviço (em kN): V_k

Ler o momento torçor de serviço (em kNm): T_k

Fim da entrada de dados e início dos cálculos.

2) Parâmetros do diagrama retangular para o concreto

- Se $f_{ck} \leq 50$ MPa:

$$\lambda = 0,8$$

$$\alpha_c = 0,85$$

$$\varepsilon_u \left(\frac{o}{oo} \right) = 3,5$$

$$\xi_{lim} = 0,8\beta - 0,35$$

- Se $f_{ck} > 50$ MPa:

$$\lambda = 0,8 - \frac{(f_{ck} - 50)}{400}$$

$$\alpha_c = 0,85 \left[1 - \frac{(f_{ck} - 50)}{200} \right]$$

$$\varepsilon_u \left(\frac{o}{oo} \right) = 2,6 + 35 \left(\frac{90 - f_{ck}}{100} \right)^4$$

$$\xi_{lim} = 0,8\beta - 0,45$$

3) Conversão de unidades para kN e cm

$$M_k = 100M_k$$

$$T_k = 100T_k$$

$$E_s = 100E_s$$

4) Resistências de cálculo em kN/cm²

$$f_{cd} = f_{ck} / (10\gamma_c)$$

$$\sigma_{cd} = \alpha_c f_{cd}$$

$$f_{yd} = f_{yk} / (10\gamma_s)$$

5) Esforços solicitantes de cálculo

$$M_d = \gamma_f M_k$$

$$V_d = \gamma_f V_k$$

$$T_d = \gamma_f T_k$$

6) Altura útil e parâmetro geométrico

$$d = h - d'$$

$$\delta = d'/h$$

7) Dimensionamento à flexão simples (Ver o capítulo 2)

$$\mu_{\text{lim}} = \lambda \xi_{\text{lim}} (1 - 0,5 \lambda \xi_{\text{lim}})$$

$$\mu = \frac{M_d}{bd^2 \sigma_{cd}}$$

- Se $\mu \leq \mu_{\text{lim}}$ (armadura simples):

$$\xi = \frac{1 - \sqrt{1 - 2\mu}}{\lambda}$$

$$A_s = \lambda \xi b d \frac{\sigma_{cd}}{f_{yd}}$$

$$A'_s = 0.$$

- Se $\mu > \mu_{\text{lim}}$ (armadura dupla):

- ✓ Evitar o uso de armadura dupla no domínio 2:

$$\xi_a = \frac{\varepsilon_u}{\varepsilon_u + 10 \text{‰}}$$

Se $\xi_{\text{lim}} < \xi_a$, colocar mensagem para o usuário aumentar as dimensões da seção transversal e parar o processamento.

- ✓ Evitar o caso em que $\xi_{\text{lim}} \leq \delta$, pois a armadura de compressão ficaria tracionada. Colocar mensagem para o usuário aumentar as dimensões da seção transversal e parar o processamento.

$$\varepsilon'_s = \varepsilon_u \left(\frac{\xi_{\text{lim}} - \delta}{\xi_{\text{lim}}} \right)$$

Chamar uma sub-rotina para calcular a tensão σ'_{sd} .

$$A'_s = \frac{(\mu - \mu_{\text{lim}}) b d \sigma_{cd}}{(1 - \delta) \sigma'_{sd}}$$

$$A_s = \left(\lambda \xi_{\text{lim}} + \frac{\mu - \mu_{\text{lim}}}{1 - \delta} \right) \frac{bd \sigma_{cd}}{f_{yd}}$$

Cálculo da armadura mínima:

$$f_{yd} = 10 f_{yd} \text{ (voltando para MPa)}$$

- Se $f_{ck} \leq 50 \text{ MPa}$:

$$\rho_{\min} = \frac{0,078 f_{ck}^{2/3}}{f_{yd}}$$

- Se $f_{ck} > 50 \text{ MPa}$:

$$\rho_{\min} = \frac{0,5512 \ln(1 + 0,11 f_{ck})}{f_{yd}}$$

Se $\rho_{\min} < 0,15\% \rightarrow$ fazer $\rho_{\min} = 0,15\%$

$$A_{s,\min} = \rho_{\min} b h$$

Se $A_s < A_{s,\min}$, adotar $A_s = A_{s,\min}$

8) Tensões devidas ao esforço cortante (Ver o Capítulo 5)

$$\tau_{wd} = \frac{V_d}{b_w d}$$

Passar para MPa: $\tau_{wd} = 10 \tau_{wd}$

$$\alpha_v = 1 - \frac{f_{ck}}{250}$$

$$\tau_{wu} = 0,27 \alpha_v f_{cd}$$

9) Seção vazada equivalente para a torção

Ver Figura 1.2.1 e Figura 1.2.2 do Volume 4 de Curso de Concreto Armado.

$$C_1 = d'$$

$$t_0 = \frac{bh}{2(b+h)}$$

- Se $t_0 \geq 2C_1$:

Espessura da parede: $t = t_0$

Área limitada pela linha média: $A_e = (b-t)(h-t)$

Perímetro da linha média: $u = 2(b+h-2t)$

- Se $t_0 < 2C_1$:

$$t_{\max} = b - 2C_1$$

Espessura da parede: $t = t_0 \leq t_{\max}$

Área limitada pela linha média: $A_e = (b - 2C_1)(h - 2C_1)$

Perímetro da linha média: $u = 2(b + h - 4C_1)$

10) Tensões devidas ao momento torçor

$$\tau_{td} = \frac{T_d}{2A_e t}$$

Passar para MPa: $\tau_{td} = 10\tau_{td}$

$$\tau_{tu} = 0,25\alpha_v f_{cd}$$

11) Verificação do esmagamento das bielas

$$Soma = \frac{\tau_{td}}{\tau_{tu}} + \frac{\tau_{wd}}{\tau_{wu}}$$

- Se $Soma > 1$

Colocar mensagem informando a ruptura das bielas, para o usuário aumentar as dimensões da seção transversal, e parar o processamento.

12) Espaçamento máximo dos estribos

- Se $Soma \leq 0,67$:

$$s_{\max} = 0,6d \leq 30 \text{ cm}$$

- Se $Soma > 0,67$:

$$s_{\max} = 0,3d \leq 20 \text{ cm}$$

13) Dimensionamento dos estribos para o esforço cortante

- Se $f_{ck} \leq 50 \text{ MPa}$:

$$\tau_c = 0,126(f_{ck})^{2/3} / \gamma_c$$

- Se $f_{ck} > 50 \text{ MPa}$:

$$\tau_c = 0,8904 \ln(1 + 0,11f_{ck}) / \gamma_c$$

$$\tau_d = 1,11(\tau_{wd} - \tau_c)$$

- Se $\tau_d < 0$: fazer $\tau_d = 0$

- Se $f_{yd} > 435$: fazer $f_{yd} = 435$

$$A_{swV} = 100b_w \frac{\tau_d}{f_{yd}}$$

14) Armaduras para o momento torçor

$$f_{yd} = f_{yd} / 10 \text{ (voltando para kN/cm}^2\text{)}$$

$$A_{swT} = \frac{100T_d}{2A_e f_{yd}}$$

$$A_{sl} = \frac{T_d u}{2A_e f_{yd}}$$

15) Armadura mínima

- Se $f_{yk} > 500$: fazer $f_{yk} = 500$

- Se $f_{ck} \leq 50 \text{ MPa}$:

$$f_{ctm} = 0,30(f_{ck})^{2/3}$$

- Se $f_{ck} > 50 \text{ MPa}$:

$$f_{ctm} = 2,12 \ln(1 + 0,11 f_{ck})$$

$$\rho_{w,\min} = 0,2 \frac{f_{ctm}}{f_{yk}}$$

16) Verificação dos estribos

$$A_{sw} = A_{swV} + 2A_{swT}$$

$$A_{sw,\min} = \rho_{w,\min} 100b_w$$

Se $A_{sw} < A_{sw,\min}$: fazer $A_{sw} = A_{sw,\min}$.

17) Verificação da armadura longitudinal da torção

$$A_{sl,\min} = \frac{\rho_{w,\min}}{2} u b_w$$

Se $A_{sl} < A_{sl,\min}$: fazer $A_{sl} = A_{sl,\min}$.

Imprimir ou mostrar os resultados.

B) Sub-rotina tensão

Parâmetros de entrada: E_s , ε_s , f_{yd}

Parâmetro de saída: σ_{sd}

1) Trabalhar com o valor absoluto da deformação

$$\varepsilon_{ss} = |\varepsilon_s|$$

2) Deformação de escoamento de cálculo do aço

$$\varepsilon_{yd} = f_{yd} / E_s$$

3) Cálculo da tensão

- Se $\varepsilon_{ss} < \varepsilon_{yd}$:

$$\sigma_{sd} = E_s \varepsilon_{ss}$$

- Se $\varepsilon_{ss} \geq \varepsilon_{yd}$:

$$\sigma_{sd} = f_{yd}$$

4) Acertando o sinal da tensão

Se $\varepsilon_s < 0$, fazer $\sigma_{sd} = -\sigma_{sd}$

Fim da sub-rotina.

8.3 – Programa em Fortran 90

A listagem a seguir apresenta o código do programa em Fortran 90. Os dados são lidos do arquivo DADOS.TXT. Para facilitar a comparação com os programas em Visual Basic e Visual C#, cada dado é lido em uma linha do arquivo. As mensagens de alerta e os resultados são mostrados na tela. Os dados devem ser fornecidos usando somente o ponto como separador decimal. O arquivo com a listagem do código-fonte em Fortran 90 pode ser obtido em www.editoradunas.com.br/programas.htm.

LISTAGEM DO CÓDIGO-FONTE EM FORTRAN 90

```
!
!Dimensionamento de seções retangulares à Torção combinada com Flexão e Esforço Cortante
!
!
!PROGRAMA PRINCIPAL
!
```

```
!Declarando variáveis reais de dupla precisão
IMPLICIT REAL*8(A-H,O-Z)
!
!Abertura do arquivo de dados
OPEN(1,FILE='DADOS.TXT')
!
!LEITURA DOS DADOS
!
!Os dados são lidos do arquivo DADOS.TXT
!Cada dado deve estar em uma linha do arquivo
!
!
!Resistência característica à compressão do concreto em MPa
read(1,*)fck
!
!Tensão de escoamento característica do aço em MPa
read(1,*)fyk
!
!Módulo de elasticidade do aço em GPa
read(1,*)es
!
!Coeficientes parciais de segurança:
!para o concreto
read(1,*)gamac
!
!para o aço
read(1,*)gamas
!
!para o momento fletor
read(1,*)gamaf
!
!Coeficiente beta de redistribuição de momentos
read(1,*)bduct
!
!Largura da seção transversal
read(1,*)b
!
!Altura da seção transversal
read(1,*)h
!
!Parâmetro d!
read(1,*)dl
!
!Momento fletor de serviço em kNm
read(1,*)amk
!
!Esforço cortante em kN
read(1,*)vk
!
!Momento torçor em kNm
read(1,*)tk
!
!FIM DA ENTRADA DE DADOS
!
!INÍCIO DOS CÁLCULOS
!
```

```

!Parâmetros do diagrama retangular
If (fck<=50) Then
  alamb=0.8
  alfac=0.85
  eu=3.5
  qlim=0.8*bduct-0.35
Else
  alamb=0.8-(fck-50)/400
  alfac=0.85*(1-(fck-50)/200)
  eu=2.6+35*((90-fck)/100)**4
  qlim=0.8*bduct-0.45
End If
!
!
!Conversão de unidades: transformando para kN e cm
amk=100*amk
tk=100*tk
es=100*es
!
!Resistências de cálculo em kN/cm2
!
fcd=fck/(10*gamac)
tcd=alfac*fcd
fyd=fyk/(10*gamas)
!
!Esforços solicitantes de cálculo
!
amd=gamaf*amk
vd=gamaf*vk
td=gamaf*tk
!
!Altura útil
d=h-dl
!
!Parâmetro geométrico
delta=dl/d
!
!
!Dimensionamento à Flexão Simples
!
!
!Momento limite
amilim=alamb*qlim*(1-0.5*alamb*qlim)
!
!Momento reduzido solicitante
ami=amd/(b*d*d*tcd)
!
If (ami<=amilim) Then
!Armadura simples
  qsi=(1- Sqrt(1-2*ami))/alamb
  aas=alamb*qsi*b*d*tcd/fyd
  asl=0
Else
!Armadura dupla
!
! Evitando armadura dupla no domínio 2
  qsia=eu/(eu+10)

```



```

If (qlim<qsia) Then
!
! Está resultando armadura dupla no domínio 2.
! Colocar mensagem para o usuário aumentar as dimensões da seção transversal e
! parar o processamento
!
  write(*,10)
10 format(//,5x,'Resultou armadura dupla no domínio 2. Aumente as dimensoes da secao transversal.')
  Goto 200
End If
!
! Eliminando o caso em que qlim<delta
! Se isto ocorrer, a armadura de compressão estará tracionada
!
  If (qlim<=delta) Then
!
! Colocar mensagem para o usuário aumentar as dimensões da seção transversal e
! parar o processamento
!
    write(*,20)
20 format(//,5x,'Aumente as dimensoes da secao transversal.')
    Goto 200
  End If
!
! Deformação da armadura de compressão
  esl=eu*(qlim-delta)/qlim
  esl=esl/1000
! Tensão na armadura de compressão
! Chamar sub-rotina
  Call Tensao(es,esl,fyd,tsl)
  asl=(ami-amilim)*b*d*tcd/((1-delta)*tsl)
  aas=(alamb*qlim+(ami-amilim)/(1-delta))*b*d*tcd/fyd
End If
!
!Armadura mínima
a=2./3.
fyd=10*fyd
If (fck<=50) Then
  romin=0.078*(fck**a)/fyd
Else
  romin=0.5512*Log(1+0.11*fck)/fyd
End If
If (romin<0.0015) Then
  romin=0.0015
End If
!
asmin=romin*b*h
If (aas<asmin) Then
  aas=asmin
End If
!
!
!Dimensionamento ao Esforço Cortante e Torção
!
!
!Verificação do esmagamento das bielas considerando a superposição das tensões
!

```

```

!Parcela devida ao esforço cortante
!
bw=b
!Tensão convencional de cisalhamento
twd=vd/(bw*d)
!Passando para MPa
twd=10*twd
!
!Tensão de cisalhamento última
fcd=10*fcd
av=1-fck/250
twu=0.27*av*fcd
!
!Parcela devida ao momento torçor
!
!Parâmetros da seção vazada equivalente
!Ver Capítulo 1 do Volume 4 de Curso de Concreto Armado
!Com referência às Figuras 1.2.1 e 1.2.2
!t0=espessura da parede da seção vazada
!ae=área limitada pela linha média
!up=perímetro da linha média
!
c1=dl
t0=b*h/(2*(b+h))
If (t0>=2*c1) Then
! Caso 1: Fig.1.2.1
t=t0
ae=(b-t)*(h-t)
up=2*(b+h-2*t)
Else
! Caso 2: Fig.1.2.2
t=t0
tmax=b-2*c1
If (t>tmax) Then
t=tmax
End If
ae=(b-2*c1)*(h-2*c1)
up=2*(b+h-4*c1)
End If
!Tensão convencional de cisalhamento
ttd=td/(2*ae*t)
!Passando para MPa
ttd=10*ttd
!Tensão de cisalhamento última
ttu=0.25*av*fcd
!
!Combinação das tensões e verificação
soma=ttd/ttu+twd/twu
If (soma> 1) Then
write(*,30)
30 format(/,5x,'Esmagamento da biela de compressão. Aumente as dimensoes da secao transversal.')
Goto 200
End If
!
!
!Espaçamento máximo dos estribos
!

```

```

If (soma<=0.67) Then
    smax=0.6*d
    If (smax>30) Then
        smax=30
    End If
Else
    smax=0.3*d
    If (smax>20) Then
        smax=20
    End If
End If
!
!Dimensionamento das armaduras para o esforço cortante
!
!
!Tensão Talc de redução da armadura
!Se gamac=1.4 irão resultar as expressões (7.4.6) e (7.4.7) do Volume 1 de Curso
!de Concreto Armado
!
If (fck<=50) Then
    a=2./3.
    tc=0.126*(fck**a)/gamac
Else
    tc=0.8904*Log(1+0.11*fck)/gamac
End If
!
!Tensão Tald para cálculo da armadura
!
tald=1.11*(twd-tc)
If (tald<0) Then
    tald=0
End If
!
!Limitação da tensão de escoamento do aço conforme a NBR-6118
!
If(fyd>435) Then
    fyd=435
End If
!
!Cálculo da armadura
aswv=100*bw*tald/fyd
!
!Dimensionamento das armaduras para o momento torçor
!
fyd=fyd/10
aswt=100*td/(2*ae*fyd)
aslt=td*up/(2*ae*fyd)
!
!
!Cálculo da armadura mínima:
!A tensão fyk deve ser menor ou igual a 500 MPa
fykmax=fyk
If (fykmax>500) Then
    fykmax=500
End If
!
!Resistência média à tração do concreto

```

```

!
If (fck<=50) Then
  a=2./3.
  fctm=0.3*(fck**a)
Else
  fctm=2.12*Log(1+0.11*fck)
End If
!
!Taxa mínima de armadura (ver equação (1.4.5) do Volume 4)
rowmin=0.2*fctm/fykmax
!
!Superposição dos estribos
asw=aswv+2*aswt
aswmin=rowmin*100*bw
If (asw<aswmin) Then
  asw=aswmin
End If
!
!Armadura longitudinal mínima da torção
!
aslmin=0.5*rowmin*up*bw
If (aslt<aslmin) Then
  aslt=aslmin
End If
!
!
!Mostrando os resultados na tela
!
write(*,40)
40 format(/,5x,'Estribos de dois ramos:',/)
write(*,50)aswv
50 format(5x,'Para o cortante =',f6.2,2x,' cm2/m')
write(*,60)aswt
60 format(5x,'Para o torcor =',f6.2,2x,' cm2/m')
write(*,70)asw
70 format(5x,'Total =',f6.2,2x,' cm2/m')
write(*,80)smax
80 format(5x,'Espacamento maximo (cm) =',f6.1,/)
write(*,90)
90 format(5x,'Armadura longitudinal para o momento fletor:')
write(*,100)aas,asl
100 format(5x,'As =',f6.2,2x,' cm2',4x,'Aslinha =',f6.2,2x,' cm2')
write(*,110)aslt
110 format(/,5x,'Armadura longitudinal para o torcor =',f6.2,2x,' cm2')
!Encerrando o programa
200 a=0
End
!
!
!
Subroutine Tensao(es, esl, fyd, tsl)
IMPLICIT REAL*8(A-H,O-Z)
!
!Calcula a tensão no aço
!es = módulo de elasticidade do aço em kN/cm2
!esl = deformação de entrada

```

```

!fyd = tensão de escoamento de cálculo em kN/cm2
!tsl = tensão de saída em kN/cm2
!
!Trabalhando com deformação positiva
ess = Abs(esl)
eyd = fyd / es
If (ess < eyd) Then
    tsl = es * ess
Else
    tsl = fyd
End If
!Trocando o sinal se necessário
If (esl < 0) Then
    tsl = -tsl
End If
Return
End

```

8.4 – Programa em Visual Basic 2017

Na fig. 8.4.1 apresenta-se o formulário do programa escrito em Visual Basic 2017. Os dados podem ser fornecidos usando o ponto ou a vírgula como separador decimal.

The form is titled "Dimensionamento à Torção". It features a diagram of a rectangular cross-section with width b , height h , effective depth d , and effective depth of the top reinforcement d' . The diagram also shows the areas of longitudinal reinforcement A_s and A'_s .

The form is organized into several sections:

- Materials:**
 - f_{ck} (MPa) =
 - f_{yk} (MPa) =
 - E_s (GPa) =
- Coefficientes parciais:**
 - γ_{mac} =
 - γ_{mas} =
 - γ_{maf} =
- Coefficiente de redistribuição:**
 - β =
- Esforços de serviço:**
 - M_k (kNm) =
 - V_k (kN) =
 - T_k (kNm) =
- Dimensões da seção:**
 - b (cm) =
 - h (cm) =
 - d' (cm) =
- Estribos de dois ramos:**
 - Para o cortante = cm²/m
 - Para o torção = cm²/m
 - Total = cm²/m
 - Espaçamento máximo (cm) =
- Armadura longitudinal (cm²):**
 - Para o momento fletor:
 - A_s =
 - A'_s =
 - Para o momento torção:
 - A_{sl} =

A "CALCULAR" button is located at the bottom right of the form.

Fig. 8.4.1 – Formulário do programa em Visual Basic 2017

Para criar o projeto, procede-se como descrito na seção 1.4 do capítulo 1. Utilizando a Caixa de Ferramentas, são introduzidos os controles no Form1. A propriedade Text do Form1 é "Dimensionamento à Torção", como aparece no topo do formulário. O arquivo com a listagem do código-fonte em Visual Basic 2017 pode ser obtido em

www.editoradunas.com.br/programas.htm. Após a elaboração do formulário, basta clicar com o botão direito do mouse sobre o formulário para Exibir Código e colar o código-fonte.

O formulário contém os seguintes controles:

Tipo	Nome	Descrição
PictureBox	PictureBox1	Imagem da seção
Label	Label1	fck (MPa)=
TextBox	TextBox1	Valor de fck
Label	Label2	fyk (MPa)=
TextBox	TextBox2	Valor de fyk
Label	Label3	Es (GPa)=
TextBox	TextBox3	Valor de Es
Label	Label4	Gamac =
TextBox	TextBox4	Valor de Gamac
Label	Label5	Gamas =
TextBox	TextBox5	Valor de Gamas
Label	Label6	Gamaf =
TextBox	TextBox6	Valor de Gamaf
Label	Label7	Beta =
TextBox	TextBox7	Valor de Beta
Label	Label8	b (cm)=
TextBox	TextBox8	Valor de b
Label	Label9	h (cm)=
TextBox	TextBox9	Valor de h
Label	Label10	d' (cm)=
TextBox	TextBox10	Valor de d'
Label	Label11	Mk (kNm)=
TextBox	TextBox11	Valor de Mk
Label	Label12	Vk (kN)=
TextBox	TextBox12	Valor de Vk
Label	Label13	Tk (kNm)=
TextBox	TextBox13	Valor de Tk
Label	Label14	Para o cortante =
TextBox	TextBox14	Área de estribos para o cortante
Label	Label15	cm ² /m
Label	Label16	Para o torçor =
TextBox	TextBox15	Área de estribos para o torçor
Label	Label17	cm ² /m
Label	Label18	Total =
TextBox	TextBox16	Área total de estribos
Label	Label19	cm ² /m
Label	Label20	Espaçamento máximo (cm)=
TextBox	TextBox17	Valor do espaçamento máximo
Label	Label21	Para o momento fletor:
Label	Label22	As =
TextBox	TextBox18	Valor de As
Label	Label23	A's =
TextBox	TextBox19	Valor de A's
Label	Label24	Para o momento torçor:
Label	Label25	Asl =
TextBox	TextBox20	Valor de Asl
Button	Button1	CALCULAR
GroupBox	GroupBox1	Materiais
GroupBox	GroupBox2	Coeficientes parciais

GroupBox	GroupBox3	Coeficiente de redistribuição
GroupBox	GroupBox4	Dimensões da seção
GroupBox	GroupBox5	Esforços de serviço
GroupBox	GroupBox6	Estribos de dois ramos
GroupBox	GroupBox7	Armadura longitudinal (cm ²)

LISTAGEM DO CÓDIGO-FONTE EM VISUAL BASIC 2017

Public Class Form1

Public WithEvents As Double

Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click

Dim fck, fyk, es, gamac, gamas, gamaf, bduct As Double

Dim b, h, d, dl, amk, vk, tk As Double

Dim alamb, alfac, eu, qlim As Double

Dim fcd, tcd, fyd, amd, vd, td As Double

Dim delta, amilim, ami, qsi, qsia, aas, asl As Double

Dim esl, a, romin, asmin As Double

Dim bw, twd, av, twu As Double

Dim c1, t0, t, ae, up, tmax As Double

Dim ttd, ttu, soma, smax As Double

Dim tc, tald, aswv, aswt, aslt As Double

Dim fykmax, fctm, rowmin, asw, aswmin, aslmin As Double

,

,

'Esta sub-rotina corresponde ao botão CALCULAR.

,

,

,

'Dimensionamento de seções retangulares à Torção combinada com Flexão e Esforço Cortante

,

'ENTRADA DE DADOS

,

'Os dados são lidos das caixas de texto do formulário

,

'Resistência característica à compressão do concreto em MPa

'Substituindo a vírgula por ponto

TextBox1.Text = Replace(TextBox1.Text, ",", ".")

fck = Val(TextBox1.Text)

,

'Tensão de escoamento característica do aço em MPa

TextBox2.Text = Replace(TextBox2.Text, ",", ".")

fyk = Val(TextBox2.Text)

,

'Módulo de elasticidade do aço em GPa

TextBox3.Text = Replace(TextBox3.Text, ",", ".")

es = Val(TextBox3.Text)

,

'Coeficientes parciais de segurança:

'para o concreto

TextBox4.Text = Replace(TextBox4.Text, ",", ".")

gamac = Val(TextBox4.Text)

,

```

'para o aço
TextBox5.Text = Replace(TextBox5.Text, ",", ".")
gamas = Val(TextBox5.Text)
,

'para o momento fletor
TextBox6.Text = Replace(TextBox6.Text, ",", ".")
gamaf = Val(TextBox6.Text)
,

'Coeficiente beta de redistribuição de momentos
TextBox7.Text = Replace(TextBox7.Text, ",", ".")
bduct = Val(TextBox7.Text)
,

'Largura da seção transversal
TextBox8.Text = Replace(TextBox8.Text, ",", ".")
b = Val(TextBox8.Text)
,

'Altura da seção transversal
TextBox9.Text = Replace(TextBox9.Text, ",", ".")
h = Val(TextBox9.Text)
,

'Parâmetro d'
TextBox10.Text = Replace(TextBox10.Text, ",", ".")
dl = Val(TextBox10.Text)
,

'Momento fletor de serviço em kNm
TextBox11.Text = Replace(TextBox11.Text, ",", ".")
amk = Val(TextBox11.Text)
,

'Esforço cortante em kN
TextBox12.Text = Replace(TextBox12.Text, ",", ".")
vk = Val(TextBox12.Text)
,

'Momento torçor em kNm
TextBox13.Text = Replace(TextBox13.Text, ",", ".")
tk = Val(TextBox13.Text)
,

'FIM DA ENTRADA DE DADOS
,

'INÍCIO DOS CÁLCULOS
,

'Parâmetros do diagrama retangular
If (fck <= 50) Then
    alamb = 0.8
    alfac = 0.85
    eu = 3.5
    qlim = 0.8 * bduct - 0.35
Else
    alamb = 0.8 - (fck - 50) / 400
    alfac = 0.85 * (1 - (fck - 50) / 200)
    eu = 2.6 + 35 * ((90 - fck) / 100) ^ 4
    qlim = 0.8 * bduct - 0.45
End If
,

'Conversão de unidades: transformando para kN e cm
amk = 100 * amk
tk = 100 * tk

```



```

es = 100 * es
,
'Resistências de cálculo em kN/cm2
,
fcd = fck / (10 * gamac)
tcd = alfac * fcd
fyd = fyk / (10 * gamas)
,
'Esforços solicitantes de cálculo
,
amd = gamaf * amk
vd = gamaf * vk
td = gamaf * tk
,
'Altura útil
d = h - dl
,
'Parâmetro geométrico
delta = dl / d
,
,
'Dimensionamento à Flexão Simples
,
,
'Momento limite
amilim = alamb * qlim * (1 - 0.5 * alamb * qlim)
,
'Momento reduzido solicitante
ami = amd / (b * d * d * tcd)
,
If (ami <= amilim) Then
    'Armadura simples
    qsi = (1 - Math.Sqrt(1 - 2 * ami)) / alamb
    aas = alamb * qsi * b * d * tcd / fyd
    asl = 0
Else
    'Armadura dupla
    ,
    'Evitando armadura dupla no domínio 2
    qsia = eu / (eu + 10)
    If (qlim < qsia) Then
        ,
        'Está resultando armadura dupla no domínio 2.
        'Colocar mensagem para o usuário aumentar as dimensões da seção transversal e parar o
        processamento
        ,
        MsgBox("Resultou armadura dupla no domínio 2. Aumente as dimensões da seção
        transversal", vbOKOnly + vbExclamation)
        Exit Sub
    End If
    ,
    'Eliminando o caso em que qlim<delta
    'Se isto ocorrer, a armadura de compressão estará tracionada
    ,
    If (qlim <= delta) Then
        ,

```

'Colocar mensagem para o usuário aumentar as dimensões da seção transversal e parar o processamento

```

    MsgBox("Aumente as dimensões da seção transversal", vbOKOnly + vbExclamation)
    Exit Sub
End If
'
'Deformação da armadura de compressão
esl = eu * (qlim - delta) / qlim
esl = esl / 1000
'Tensão na armadura de compressão
'Chamar sub-rotina
Call Tensao(es, esl, fyd)
asl = (ami - amilim) * b * d * tcd / ((1 - delta) * tsl)
aas = (alamb * qlim + (ami - amilim) / (1 - delta)) * b * d * tcd / fyd
End If
'
'Armadura mínima
a = 2 / 3
fyd = 10 * fyd
If (fck <= 50) Then
    romin = 0.078 * (fck ^ a) / fyd
Else
    romin = 0.5512 * Math.Log(1 + 0.11 * fck) / fyd
End If
If (romin < 0.0015) Then
    romin = 0.0015
End If
'
asmin = romin * b * h
If (aas < asmin) Then
    aas = asmin
End If
'
'
'Dimensionamento ao Esforço Cortante e Torção
'
'
'Verificação do esmagamento das bielas considerando a superposição das tensões
'
'Parcela devida ao esforço cortante
'
bw = b
'Tensão convencional de cisalhamento
twd = vd / (bw * d)
'Passando para MPa
twd = 10 * twd
'
'Tensão de cisalhamento última
fcd = 10 * fcd
av = 1 - fck / 250
twu = 0.27 * av * fcd
'
'Parcela devida ao momento torçor
'
'Parâmetros da seção vazada equivalente
'Ver Capítulo 1 do Volume 4 de Curso de Concreto Armado

```

```

'Com referência às Figuras 1.2.1 e 1.2.2
't0=espessura da parede da seção vazada
'ae=área limitada pela linha média
'up=perímetro da linha média
,

c1 = dl
t0 = b * h / (2 * (b + h))
If (t0 >= 2 * c1) Then
    ' Caso 1: Fig.1.2.1
    t = t0
    ae = (b - t) * (h - t)
    up = 2 * (b + h - 2 * t)
Else
    ' Caso 2: Fig.1.2.2
    t = t0
    tmax = b - 2 * c1
    If (t > tmax) Then
        t = tmax
    End If
    ae = (b - 2 * c1) * (h - 2 * c1)
    up = 2 * (b + h - 4 * c1)
End If
'Tensão convencional de cisalhamento
ttd = td / (2 * ae * t)
'Passando para MPa
ttd = 10 * ttd
'Tensão de cisalhamento última
ttu = 0.25 * av * fcd
,

'Combinação das tensões e verificação
soma = ttd / ttu + twd / twu
If (soma > 1) Then
    MsgBox("Esmagamento da biela de compressão. Aumente as dimensões da seção transversal.", vbOKOnly + vbExclamation)
    Exit Sub
End If
,
,

'Espaçamento máximo dos estribos
,

If (soma <= 0.67) Then
    smax = 0.6 * d
    If (smax > 30) Then
        smax = 30
    End If
Else
    smax = 0.3 * d
    If (smax > 20) Then
        smax = 20
    End If
End If
,

'Dimensionamento das armaduras para o esforço cortante
,
,

'Tensão Talc de redução da armadura

```

'Se $\gamma_{mac}=1.4$ irão resultar as expressões (7.4.6) e (7.4.7) do Volume 1 de Curso de Concreto Armado

```

If (fck <= 50) Then
  a = 2 / 3
  tc = 0.126 * (fck ^ a) / gamac
Else
  tc = 0.8904 * Math.Log(1 + 0.11 * fck) / gamac
End If

```

'Tensão Tald para cálculo da armadura

```

tald = 1.11 * (twd - tc)
If (tald < 0) Then
  tald = 0
End If

```

'Limitação da tensão de escoamento do aço conforme a NBR-6118

```

If (fyd > 435) Then
  fyd = 435
End If

```

'Cálculo da armadura

```
aswv = 100 * bw * tald / fyd
```

'Dimensionamento das armaduras para o momento torçor

```

fyd = fyd / 10
aswt = 100 * td / (2 * ae * fyd)
aslt = td * up / (2 * ae * fyd)

```

'Cálculo da armadura mínima:

'A tensão f_{yk} deve ser menor ou igual a 500 MPa

```

fykmax = fyk
If (fykmax > 500) Then
  fykmax = 500
End If

```

'Resistência média à tração do concreto

```

If (fck <= 50) Then
  a = 2 / 3
  fctm = 0.3 * (fck ^ a)
Else
  fctm = 2.12 * Math.Log(1 + 0.11 * fck)
End If

```

'Taxa mínima de armadura (ver equação (1.4.5) do Volume 4)

```
rowmin = 0.2 * fctm / fykmax
```

'Superposição dos estribos

```

asw = aswv + 2 * aswt
aswmin = rowmin * 100 * bw
If (asw < aswmin) Then
  asw = aswmin

```

```

End If
'
'Armadura longitudinal mínima da torção
'
aslmin = 0.5 * rowmin * up * bw
If (aslt < aslmin) Then
    aslt = aslmin
End If
'
'
'Convertendo as saída para duas casas decimais
aswv = FormatNumber(aswv, 2)
aswt = FormatNumber(aswt, 2)
asw = FormatNumber(asw, 2)
aas = FormatNumber(aas, 2)
asl = FormatNumber(asl, 2)
aslt = FormatNumber(aslt, 2)
'Convertendo o espaçamento máximo dos estribos para
'uma casa decimal
smax = FormatNumber(smax, 1)
'
'Mostrando os resultados nas caixas de texto
'
TextBox14.Text = aswv
TextBox15.Text = aswt
TextBox16.Text = asw
TextBox17.Text = smax
TextBox18.Text = aas
TextBox19.Text = asl
TextBox20.Text = aslt
'
End Sub
Private Sub Tensao(es, esl, fyd)
    Dim ess, eyd As Double
    '
    'Calcula a tensão no aço
    'es = módulo de elasticidade do aço em kN/cm2
    'esl = deformação de entrada
    'fyd = tensão de escoamento de cálculo em kN/cm2
    'tsl = tensão de saída em kN/cm2
    '
    'Trabalhando com deformação positiva
    ess = Math.Abs(esl)
    eyd = fyd / es
    If (ess < eyd) Then
        tsl = es * ess
    Else
        tsl = fyd
    End If
    'Trocando o sinal se necessário
    If (esl < 0) Then
        tsl = -tsl
    End If
End Sub
End Class

```

8.5 – Programa em Visual C# 2017

Na fig. 8.5.1 apresenta-se o formulário do programa escrito em Visual C# 2017. O formulário é inteiramente análogo ao do programa em Visual Basic 2017. Os dados podem ser fornecidos usando o ponto ou a vírgula como separador decimal. Os controles são os mesmos do programa em Visual Basic.

O arquivo com a listagem do código-fonte em Visual C# 2017 pode ser obtido em www.editoradunas.com.br/programas.htm. Após a elaboração do formulário, basta clicar com o botão direito do mouse sobre o formulário para Exibir Código e colar o código-fonte.

Fig. 8.5.1 – Formulário do programa em Visual C# 2017

LISTAGEM DO CÓDIGO-FONTE EM VISUAL C# 2017

```
using System;
using System.Globalization;
using System.Windows.Forms;

namespace TORQUE
{
    public partial class Form1 : Form
    {
        //Declarando variáveis públicas
        public double es, esl, fyd, tsl;
        public double aas, asl;

        public Form1()
        {
```

```

InitializeComponent();
}

private void button1_Click(object sender, EventArgs e)
{
    double fck, fyk, gamac, gamas, gamaf, bduct;
    double b, h, d, dl, amk, vk, tk;
    double alamb, alfac, eu, qlim;
    double fcd, tcd, amd, vd, td;
    double delta, amilim, ami, qsi, qsia;
    double a, romin, asmin;
    double bw, twd, av, twu;
    double c1, t0, t, ae, up, tmax;
    double ttd, ttu, soma, smax;
    double tc, tald, aswv, aswt, aslt;
    double fykmax, fctm, rowmin, asw, aswmin, aslmin;
    //
    //
    //Esta sub-rotina corresponde ao botão CALCULAR.
    //
    //
    //
    //Dimensionamento de seções retangulares à Torção combinada com Flexão e Esforço Cortante
    //
    //ENTRADA DE DADOS
    //
    //Os dados são lidos das caixas de texto do formulário
    //
    //Resistência característica à compressão do concreto em MPa
    //Substituindo a vírgula por ponto
    //Resistência característica à compressão do concreto em MPa
    //Substituindo a vírgula por ponto
    textBox1.Text = textBox1.Text.Replace(",", ".");
    fck = Convert.ToDouble(textBox1.Text, CultureInfo.InvariantCulture);
    //
    //Tensão de escoamento característica do aço em MPa
    textBox2.Text = textBox2.Text.Replace(",", ".");
    fyk = Convert.ToDouble(textBox2.Text, CultureInfo.InvariantCulture);
    //
    //Módulo de elasticidade do aço em GPa
    textBox3.Text = textBox3.Text.Replace(",", ".");
    es = Convert.ToDouble(textBox3.Text, CultureInfo.InvariantCulture);
    //
    //Coeficientes parciais de segurança:
    //para o concreto
    textBox4.Text = textBox4.Text.Replace(",", ".");
    gamac = Convert.ToDouble(textBox4.Text, CultureInfo.InvariantCulture);
    //
    //para o aço
    textBox5.Text = textBox5.Text.Replace(",", ".");
    gamas = Convert.ToDouble(textBox5.Text, CultureInfo.InvariantCulture);
    //
    //para o momento fletor
    textBox6.Text = textBox6.Text.Replace(",", ".");
    gamaf = Convert.ToDouble(textBox6.Text, CultureInfo.InvariantCulture);
    //
    //Coeficiente beta de redistribuição de

```

```

//momentos
textBox7.Text = textBox7.Text.Replace(",", ".");
bduct = Convert.ToDouble(textBox7.Text, CultureInfo.InvariantCulture);
//
//Largura da seção transversal
textBox8.Text = textBox8.Text.Replace(",", ".");
b = Convert.ToDouble(textBox8.Text, CultureInfo.InvariantCulture);
//
//Altura da seção transversal
textBox9.Text = textBox9.Text.Replace(",", ".");
h = Convert.ToDouble(textBox9.Text, CultureInfo.InvariantCulture);
//
//Parâmetro d'
textBox10.Text = textBox10.Text.Replace(",", ".");
dl = Convert.ToDouble(textBox10.Text, CultureInfo.InvariantCulture);
//
//Momento fletor de serviço em kNm
textBox11.Text = textBox11.Text.Replace(",", ".");
amk = Convert.ToDouble(textBox11.Text, CultureInfo.InvariantCulture);
//
//Esforço cortante em kN
textBox12.Text = textBox12.Text.Replace(",", ".");
vk = Convert.ToDouble(textBox12.Text, CultureInfo.InvariantCulture);
//
//Momento torçor em kNm
textBox13.Text = textBox13.Text.Replace(",", ".");
tk = Convert.ToDouble(textBox13.Text, CultureInfo.InvariantCulture);
//
//FIM DA ENTRADA DE DADOS
//
//INÍCIO DOS CÁLCULOS
//
//Parâmetros do diagrama retangular
if (fck <= 50)
{
    alamb = 0.8;
    alfac = 0.85;
    eu = 3.5;
    qlim = 0.8 * bduct - 0.35;
}
else
{
    alamb = 0.8 - (fck - 50) / 400;
    alfac = 0.85 * (1 - (fck - 50) / 200);
    a = (90 - fck) / 100;
    eu = 2.6 + 35 * Math.Pow(a, 4);
    qlim = 0.8 * bduct - 0.45;
}
//
//Conversão de unidades: transformando para kN e cm
amk = 100 * amk;
tk = 100 * tk;
es = 100 * es;
//
//Resistências de cálculo em kN/cm2
//
fcd = fck / (10 * gamac);

```



```

tcd = alfac * fcd;
fyd = fyk / (10 * gamas);
//
//Esforços solicitantes de cálculo
//
amd = gamaf * amk;
vd = gamaf * vk;
td = gamaf * tk;
//
//Altura útil
d = h - dl;
//
//Parâmetro geométrico
delta = dl / d;
//
//
//Dimensionamento à Flexão Simples
//
//
//Momento limite
amilim = alamb * qlim * (1 - 0.5 * alamb * qlim);
//
//Momento reduzido solicitante
ami = amd / (b * d * d * tcd);
//
if (ami <= amilim)
{
    //Armadura simples
    qsi = (1 - Math.Sqrt(1 - 2 * ami)) / alamb;
    aas = alamb * qsi * b * d * tcd / fyd;
    asl = 0;
}
if (ami > amilim)
{
    //Armadura dupla
    //
    //Evitando armadura dupla no domínio 2
    qsia = eu / (eu + 10);
    if (qlim < qsia)
    {
        //
        // Está resultando armadura dupla no domínio 2.
        // Colocar mensagem para o usuário aumentar as dimensões da seção transversal e parar o
processamento
        //
        MessageBox.Show("Resultou armadura dupla no domínio 2. Aumente as dimensões da
seção transversal");
        return;
    }
    //
    // Eliminando o caso em que qlim<delta
    // Se isto ocorrer, a armadura de compressão estará tracionada
    //
    if (qlim <= delta)
    {
        //

```

```

        // Colocar mensagem para o usuário aumentar as dimensões da seção transversal e parar o
processamento
        //
        MessageBox.Show("Aumente as dimensões da seção transversal");
        return;
    }
    //
    //Deformação da armadura de compressão
    esl = eu * (qlim - delta) / qlim;
    esl = esl / 1000;
    // Tensão na armadura de compressão
    // Chamar Sub-rotina
    Tensao();
    asl = (ami - amilim) * b * d * tcd / ((1 - delta) * tsl);
    aas = (alamb * qlim + (ami - amilim) / (1 - delta)) * b * d * tcd / fyd;
}
//
//Armadura mínima
a = 2.0 / 3.0;
fyd = 10 * fyd;
if (fck <= 50)
{
    romin = 0.078 * Math.Pow(fck, a) / fyd;
}
else
{
    romin = 0.5512 * Math.Log(1 + 0.11 * fck) / fyd;
}
if (romin < 0.0015)
{
    romin = 0.0015;
}
//
asmin = romin * b * h;
if (aas < asmin)
{
    aas = asmin;
}
//
//
//Dimensionamento ao Esforço Cortante e Torção
//
//
//Verificação do esmagamento das bielas considerando a superposição das tensões
//
//Parcela devida ao esforço cortante
//
bw = b;
//Tensão convencional de cisalhamento]
twd = vd / (bw * d);
//Passando para MPa
twd = 10 * twd;
//
//Tensão de cisalhamento última
fcd = 10 * fcd;
av = 1 - fck / 250;
twu = 0.27 * av * fcd;

```

```

//
//Parcela devida ao momento torçor
//
//Parâmetros da seção vazada equivalente
//Ver Capítulo 1 do Volume 4 de Curso de Concreto Armado
//Com referência às Figuras 1.2.1 e 1.2.2
//t0=espessura da parede da seção vazada
//ae=área limitada pela linha média
//up=perímetro da linha média
//
c1 = dl;
t0 = b * h / (2 * (b + h));
if(t0 >= 2 * c1)
{
    // Caso 1: Fig.1.2.1
    t = t0;
    ae = (b - t) * (h - t);
    up = 2 * (b + h - 2 * t);
}
else
{
    // Caso 2: Fig.1.2.2
    t = t0;
    tmax = b - 2 * c1;
    if(t > tmax)
    {
        t = tmax;
    }
}
ae = (b - 2 * c1) * (h - 2 * c1);
up = 2 * (b + h - 4 * c1);
//Tensão convencional de cisalhamento
ttd = td / (2 * ae * t);
//Passando para MPa
ttd = 10 * ttd;
//Tensão de cisalhamento última
ttu = 0.25 * av * fcd;
//
//Combinação das tensões e verificação
soma = ttd / ttu + twd / twu;
if (soma > 1)
{
    MessageBox.Show("Esmagamento da biela de compressão. Aumente as dimensões da seção transversal.");
    return;
}
//
//
//Espaçamento máximo dos estribos
//
if(soma <= 0.67)
{
    smax = 0.6 * d;
    if(smax > 30)
    {
        smax = 30;
    }
}

```

```

    }
else
{
    smax = 0.3 * d;
    if(smax > 20)
    {
        smax = 20;
    }
}
//
//Dimensionamento das armaduras para o esforço cortante
//
//
//Tensão Talc de redução da armadura
//Se gamac=1.4 irão resultar as expressões (7.4.6) e (7.4.7) do Volume 1 de Curso de Concreto
Armado
//
if(fck <= 50)
{
    a = 2.0 / 3.0;
    tc = 0.126 * Math.Pow(fck, a) / gamac;
}
else
{
    tc = 0.8904 * Math.Log(1 + 0.11 * fck) / gamac;
}
//
//Tensão Tald para cálculo da armadura
//
tald = 1.11 * (twd - tc);
if(tald < 0)
{
    tald = 0;
}
//
//Limitação da tensão de escoamento do aço conforme a NBR-6118
//
if(fyd > 435)
{
    fyd = 435;
}
//
//Cálculo da armadura
aswv = 100 * bw * tald / fyd;
//
//Dimensionamento das armaduras para o momento torçor
//
fyd = fyd / 10;
aswt = 100 * td / (2 * ae * fyd);
aslt = td * up / (2 * ae * fyd);
//
//
//Cálculo da armadura mínima:
//A tensão fyk deve ser menor ou igual a 500 MPa
fykmax = fyk;
if(fykmax > 500)
{

```

```

    fykmax = 500;
}
//
//Resistência média à tração do concreto
//
if(fck <= 50)
{
    a = 2.0 / 3.0;
    fctm = 0.3 * Math.Pow(fck, a);
}
else
{
    fctm = 2.12 * Math.Log(1 + 0.11 * fck);
}
//
//Taxa mínima de armadura (ver equação (1.4.5) do Volume 4)
rowmin = 0.2 * fctm / fykmax;
//
//Superposição dos estribos
asw = aswv + 2 * aswt;
aswmin = rowmin * 100 * bw;
if(asw < aswmin)
{
    asw = aswmin;
}
//
//Armadura longitudinal mínima da torção
//
aslmin = 0.5 * rowmin * up * bw;
if(aslt < aslmin)
{
    aslt = aslmin;
}
//
//
//
//Convertendo as saída para duas casas decimais
//
decimal saida1 = Decimal.Round(Convert.ToDecimal(aswv), 2);
decimal saida2 = Decimal.Round(Convert.ToDecimal(aswt), 2);
decimal saida3 = Decimal.Round(Convert.ToDecimal(asw), 2);
decimal saida4 = Decimal.Round(Convert.ToDecimal(aas), 2);
decimal saida5 = Decimal.Round(Convert.ToDecimal(asl), 2);
decimal saida6 = Decimal.Round(Convert.ToDecimal(aslt), 2);
//Convertendo o espaçamento máximo dos estribos para uma casa decimal
decimal saida7 = Decimal.Round(Convert.ToDecimal(smax), 1);
//
//Mostrando os resultados nas caixas de texto
//
textBox14.Text = Convert.ToString(saida1);
textBox15.Text = Convert.ToString(saida2);
textBox16.Text = Convert.ToString(saida3);
textBox17.Text = Convert.ToString(saida7);
textBox18.Text = Convert.ToString(saida4);
textBox19.Text = Convert.ToString(saida5);
textBox20.Text = Convert.ToString(saida6);
//

```

```

}
private void Tensao()
{
    double ess, eyd;
    //
    //Calcula a tensão no aço
    //es = módulo de elasticidade do aço em kN/cm2
    //esl = deformação de entrada
    //fyd = tensão de escoamento de cálculo em kN/cm2
    //tsl = tensão de saída em kN/cm2
    //
    //Trabalhando com deformação positiva
    ess = Math.Abs(esl);
    eyd = fyd / es;
    if (ess < eyd)
    {
        tsl = es * ess;
    }
    else
    {
        tsl = fyd;
    }
    //Trocando o sinal se necessário
    if (esl < 0)
    {
        tsl = -tsl;
    }
}
}
}

```

8.6 – Exemplos para testar os programas

Os resultados apresentados a seguir servem para o leitor testar os programas. O dimensionamento à torção é feito para a seção da fig. 8.6.1. Esse exemplo foi resolvido manualmente no Capítulo 1 do Volume 4.

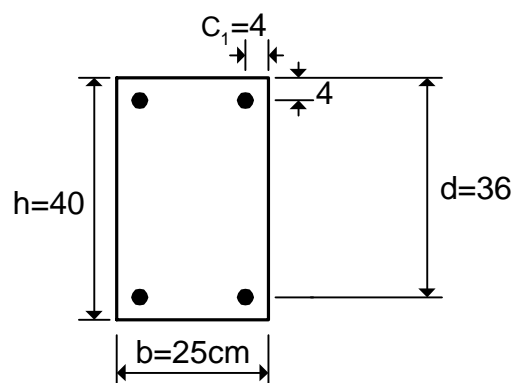


Fig. 8.6.1 - Seção transversal

São dados do problema:

$$f_{ck} = 20 \text{ MPa}; \quad f_{yk} = 500 \text{ MPa}; \quad E_s = 200 \text{ GPa}$$

$$\gamma_c = 1,4; \gamma_s = 1,15; \gamma_f = 1,4; \beta = 1$$

$$M_k = 11,03 \text{ kNm}; V_k = 17,36 \text{ kNm}; T_k = 9,60 \text{ kNm}$$

Resultados:

Estribos de dois ramos:

Para o esforço cortante: $0,00 \text{ cm}^2/\text{m}$ ($= A_{swV}$)

Para o momento torçor: $2,84 \text{ cm}^2/\text{m}$ ($= A_{swT}$)

Total: $5,68 \text{ cm}^2/\text{m}$ ($= A_{swV} + 2A_{swT}$)

Espaçamento máximo: $21,6 \text{ cm}$

Armadura longitudinal:

Para o momento fletor: $A_s = 1,5 \text{ cm}^2$; $A'_s = 0$

Para o momento torçor: $A_{sl} = 2,78 \text{ cm}^2$

Observações:

1) O programa fornece o espaçamento máximo permitido. De posse da área total de estribos (em cm^2/m), pode-se entrar na Tabela A3.3 do Apêndice 3 do Volume 2 e ler o espaçamento necessário. Se resultar um espaçamento maior que o máximo, adota-se o máximo. Evidentemente, essa parte do cálculo pode ser facilmente implementada nos programas, evitando-se o uso de tabelas.

2) Os programas podem ser utilizados para resolver os problemas de dimensionamento à flexão simples (do Capítulo 2) e ao esforço cortante (do Capítulo 5). Para isto, basta fazer dois esforços nulos e entrar com o único esforço solicitante desejado. Esses exemplos também servem para testar os programas.

Capítulo 9

DIMENSIONAMENTO À FLEXO-TRAÇÃO NORMAL

9.1 – Apresentação do problema

O objetivo é realizar o dimensionamento de uma seção retangular submetida à flexo-tração normal. A seção possui armaduras assimétricas, como indicado na fig. 9.1.1. A teoria é apresentada no Capítulo 2 do Volume 4.

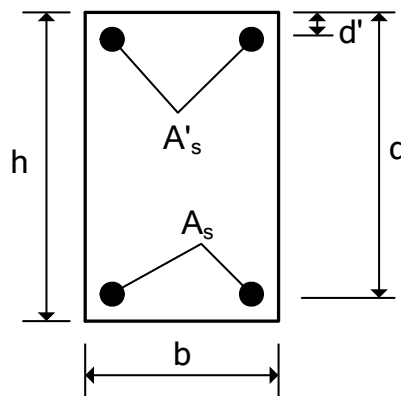


Fig. 9.1.1 - Seção transversal com armaduras assimétricas

Os dados do problema são os seguintes:

- a) propriedades dos materiais: f_{ck} , f_{yk} , E_s
- b) coeficientes parciais de segurança: γ_c , γ_s , γ_f
- c) coeficiente de redistribuição de momentos: $0,75 \leq \beta \leq 1$
- d) dimensões da seção transversal: b , h , d , d'
- e) momento fletor de serviço: M_k
- f) esforço normal de serviço: N_k

Os valores requeridos são as áreas das armaduras: A_s e A'_s .

9.2 – Roteiro para programação

A) Programa principal

1) Entrada de dados

Ler as propriedades dos materiais: f_{ck} (MPa), f_{yk} (MPa), E_s (GPa)

Ler os coeficientes parciais de segurança: γ_c , γ_s , γ_f

Ler o coeficiente de redistribuição de momentos: β

Ler as dimensões da seção (em cm): b , h , d , d'

Ler o momento fletor de serviço (em kNm): M_k

Ler o esforço normal de serviço (em kN): N_k

Fim da entrada de dados e início dos cálculos.

2) Parâmetros do diagrama retangular para o concreto

- Se $f_{ck} \leq 50$ MPa:

$$\lambda = 0,8$$

$$\alpha_c = 0,85$$

$$\varepsilon_u \left(\frac{o}{oo} \right) = 3,5$$

$$\xi_{lim} = 0,8\beta - 0,35$$

- Se $f_{ck} > 50$ MPa:

$$\lambda = 0,8 - \frac{(f_{ck} - 50)}{400}$$

$$\alpha_c = 0,85 \left[1 - \frac{(f_{ck} - 50)}{200} \right]$$

$$\varepsilon_u \left(\frac{o}{oo} \right) = 2,6 + 35 \left(\frac{90 - f_{ck}}{100} \right)^4$$

$$\xi_{lim} = 0,8\beta - 0,45$$

3) Conversão de unidades para kN e cm

$$M_k = 100M_k$$

$$f_{ck} = f_{ck} / 10$$

$$f_{yk} = f_{yk} / 10$$

$$E_s = 100E_s$$

4) Resistências de cálculo em kN/cm²

$$f_{cd} = f_{ck} / \gamma_c$$

$$\sigma_{cd} = \alpha_c f_{cd}$$

$$f_{yd} = f_{yk} / \gamma_s$$

5) Esforços solicitantes de cálculo

$$M_d = \gamma_f M_k$$

$$N_d = \gamma_f N_k$$

6) Parâmetro geométrico e esforços reduzidos

$$\delta = d'/h$$

$$\mu = \frac{M_d}{bd^2\sigma_{cd}}$$

$$\nu = \frac{N_d}{bd\sigma_{cd}}$$

7) Momento reduzido correspondente ao final do domínio 1

$$\mu_0 = 0,5(1 - \delta)\nu$$

8) Solução no domínio 1

- Se $\mu \leq \mu_0$:

Criar variável de controle: $idom = 1$

$$\omega' = \frac{0,5(1 - \delta)\nu - \mu}{(1 - \delta)}$$

$$\omega = \frac{0,5(1 - \delta)\nu + \mu}{(1 - \delta)}$$

9) Solução no domínio 2 e 3

- Se $\mu > \mu_0$:

Criar variável de controle: $idom = 2$

Calcular o momento equivalente: $\mu_{sd} = \mu - \mu_0$

Calcular o momento limite: $\mu_{lim} = \lambda\xi_{lim}(1 - 0,5\lambda\xi_{lim})$

- ✓ Se $\mu_{sd} \leq \mu_{lim}$ (armadura simples):

$$\xi = \frac{1 - \sqrt{1 - 2\mu_{sd}}}{\lambda}$$

$$\omega = \lambda\xi + \nu$$

$$\omega' = 0$$

- ✓ Se $\mu_{sd} > \mu_{lim}$ (armadura dupla):

- ✓ Evitar o uso de armadura dupla no domínio 2:

$$\xi_a = \frac{\varepsilon_u}{\varepsilon_u + 10 \frac{0}{00}}$$

Se $\xi_{\text{lim}} < \xi_a$, colocar mensagem para o usuário aumentar as dimensões da seção transversal e parar o processamento.

- ✓ Evitar o caso em que $\xi_{\text{lim}} \leq \delta$, pois a armadura de compressão ficaria tracionada. Colocar mensagem para o usuário aumentar as dimensões da seção transversal e parar o processamento.

$$\varepsilon'_s = \varepsilon_u \left(\frac{\xi_{\text{lim}} - \delta}{\xi_{\text{lim}}} \right)$$

Chamar uma sub-rotina para calcular a tensão σ'_{sd} .

$$\omega' = \frac{(\mu_{sd} - \mu_{\text{lim}}) f_{yd}}{(1 - \delta) \sigma'_{sd}}$$

$$\omega = \lambda \xi_{\text{lim}} + \frac{\mu_{sd} - \mu_{\text{lim}}}{1 - \delta} + \nu$$

10) Áreas das armaduras calculadas

$$A_s = \omega b d \frac{\sigma_{cd}}{f_{yd}}$$

$$A'_s = \omega' b d \frac{\sigma_{cd}}{f_{yd}}$$

11) Armaduras mínimas

Voltar f_{ck} e f_{yd} para MPa:

$$f_{ck} = 10 f_{ck}$$

$$f_{yd} = 10 f_{yd}$$

Caso de tração simples:

- Se $f_{ck} \leq 50$ MPa:

$$\rho_{1,\text{min}} = \frac{0,39 f_{ck}^{2/3}}{f_{yd}}$$

- Se $f_{ck} > 50$ MPa:

$$\rho_{1,\text{min}} = \frac{2,756 \ln(1 + 0,11 f_{ck})}{f_{yd}}$$

Caso de flexão simples:

- Se $f_{ck} \leq 50 \text{ MPa}$:

$$\rho_{2,\min} = \frac{0,078 f_{ck}^{2/3}}{f_{yd}}$$

- Se $f_{ck} > 50 \text{ MPa}$:

$$\rho_{2,\min} = \frac{0,5512 \ln(1 + 0,11 f_{ck})}{f_{yd}}$$

Se $\rho_{2,\min} < 0,15\% \rightarrow \rho_{2,\min} = 0,15\%$

- Se $idom = 2$:

$$\rho_{\min} = \rho_{2,\min}$$

- Se $idom = 1$:

$$\rho_{\min} = \rho_{2,\min} + (\rho_{1,\min} - \rho_{2,\min})(\mu_0 - \mu)/\mu_0$$

(interpolação linear entre $\rho_{1,\min}$ e $\rho_{2,\min}$ conforme a figura 2.4.1 do Capítulo 2 do Volume 4 de Curso de Concreto Armado)

$$A_{s,\min} = \rho_{\min} b h$$

12) Verificação da armadura mínima

- Se $idom = 2$:

Se $A_s < A_{s,\min}$, adotar $A_s = A_{s,\min}$

- Se $idom = 1$:

$$A_{s,tot} = A_s + A'_s$$

Se $A_{s,tot} < A_{s,\min}$, adotar:

$$A_s = A_s (A_{s,\min} / A_{s,tot})$$

$$A'_s = A'_s (A_{s,\min} / A_{s,tot})$$

B) Sub-rotina tensão

Parâmetros de entrada: E_s , ε_s , f_{yd}

Parâmetro de saída: σ_{sd}

1) Trabalhar com o valor absoluto da deformação

$$\varepsilon_{ss} = |\varepsilon_s|$$

2) Deformação de escoamento de cálculo do aço

$$\varepsilon_{yd} = f_{yd} / E_s$$

3) Cálculo da tensão

- Se $\varepsilon_{ss} < \varepsilon_{yd}$:

$$\sigma_{sd} = E_s \varepsilon_{ss}$$

- Se $\varepsilon_{ss} \geq \varepsilon_{yd}$:

$$\sigma_{sd} = f_{yd}$$

4) Acertando o sinal da tensão

Se $\varepsilon_s < 0$, fazer $\sigma_{sd} = -\sigma_{sd}$

Fim da sub-rotina.

9.3 – Programa em Fortran 90

A listagem a seguir apresenta o código do programa em Fortran 90. Os dados são lidos do arquivo DADOS.TXT. Para facilitar a comparação com os programas em Visual Basic e Visual C#, cada dado é lido em uma linha do arquivo. As mensagens de alerta e os resultados são mostrados na tela. Os dados devem ser fornecidos usando somente o ponto como separador decimal. O arquivo com a listagem do código-fonte em Fortran 90 pode ser obtido em www.editoradunas.com.br/programas.htm.

LISTAGEM DO CÓDIGO-FONTE EM FORTRAN 90

```
!
!Dimensionamento de seções retangulares à Flexo-Tração Normal com Armaduras Assimétricas
!
!PROGRAMA PRINCIPAL
!
!Declarando variáveis reais de dupla precisão
IMPLICIT REAL*8(A-H,O-Z)
!
!Abertura do arquivo de dados
OPEN(1,FILE='DADOS.TXT')
!
!LEITURA DOS DADOS
!
!Os dados são lidos do arquivo DADOS.TXT
!Cada dado deve estar em uma linha do arquivo
```

```
!Resistência característica à compressão do concreto em MPa
read(1,*)fck
!
!Tensão de escoamento característica do aço em MPa
read(1,*)fyk
!
!Módulo de elasticidade do aço em GPa
read(1,*)es
!
!Coeficientes parciais de segurança:
!para o concreto
read(1,*)gamac
!
!para o aço
read(1,*)gamas
!
!para o momento fletor
read(1,*)gamaf
!
!Coeficiente beta de redistribuição de momentos
read(1,*)bduct
!
!Largura da seção transversal
read(1,*)b
!
!Altura da seção transversal
read(1,*)h
!
!Altura útil
read(1,*)d
!
!Parâmetro d' (distância da armadura A's até a borda da seção)
read(1,*)dl
!
!Momento fletor de serviço em kNm
read(1,*)amk
!
!Esforço normal de serviço em kN
read(1,*)ank
!
!FIM DA ENTRADA DE DADOS
!
!INÍCIO DOS CÁLCULOS
!
!Parâmetros do diagrama retangular
If (fck<=50) Then
    alamb=0.8
    alfac=0.85
    eu=3.5
    qlim=0.8*bduct-0.35
Else
    alamb=0.8-(fck-50)/400
    alfac=0.85*(1-(fck-50)/200)
    eu=2.6+35*((90-fck)/100)**4
    qlim=0.8*bduct-0.45
End If
!
```

```

!Conversão de unidades: transformando para kN e cm
amk=100*amk
fck=fck/10
fyk=fyk/10
es=100*es
!
!Resistências de cálculo
fcd=fck/gamac
tcd=alfac*fcd
fyd=fyk/gamas
!
!Esforços solicitantes de cálculo
amd=gamaf*amk
aand=gamaf*ank
!
!Parâmetro geométrico
delta=dl/d
!
!Esforços solicitantes reduzidos
ani=aand/(b*d*tcd)
ami=amd/(b*d*d*tcd)
!
!Realização do dimensionamento
!
!Momento reduzido que define o final do domínio 1
!
ami0=0.5*(1-delta)*ani
!
If (ami<=ami0) Then
! O problema será resolvido no domínio 1
! Trata-se de flexo-tração com pequena excentricidade
! Variável para indicar o domínio
idom=1
!
! Taxas mecânicas de armadura
wl=(ami0-ami)/(1-delta)
w=(ami0+ami)/(1-delta)
End If
!
If (ami>ami0) Then
! O problema será resolvido nos domínios 2 e 3
! Trata-se de flexo-tração com grande excentricidade
! Variável para indicar o domínio
idom=2
!
! Momento equivalente
amisd=ami-ami0
! Momento limite
amilim=alamb*qlim*(1-0.5*alamb*qlim)
!
If (amisd<=amilim) Then
! Armadura simples
qsi=(1-Sqrt(1-2*amisd))/alamb
w=alamb*qsi+ani
wl=0
Else
! Armadura dupla

```

```

!   Evitando armadura dupla no domínio 2
    qsia=eu/(eu+10)
    If (qlim<qsia) Then
!
!   Está resultando armadura dupla no domínio 2.
!       Colocar mensagem para o usuário aumentar as dimensões da seção transversal e parar o
processamento
!
        write(*,10)
10    format(//,5x,'Resultou armadura dupla no domínio 2. Aumente as dimensoes da secao
transversal.')
        Goto 100
    End If
!
!   Eliminando o caso em que qlim<delta
!   Se isto ocorrer, a armadura de compressão estará tracionada
!
    If (qlim<=delta) Then
!
!       Colocar mensagem para o usuário aumentar as dimensões da seção transversal e parar o
processamento
!
        write(*,20)
20    format(//,5x,'Aumente as dimensoes da secao transversal.')
        Goto 100
    End If
!
!   Deformação da armadura de compressão
    esl=eu*(qlim-delta)/qlim
    esl=esl/1000
!   Tensão na armadura de compressão
!   Chamar sub-rotina
    Call Tensao(es,esl,fyd,tsl)
!
!   Taxas mecânicas de armadura
    wl=(amisd-amilim)*fyd/((1-delta)*tsl)
    w=alamb*qlim+(amisd-amilim)/(1-delta)+ani
    End If
End If
!
!Áreas das armaduras
aas=w*b*d*tcd/fyd
asl=wl*b*d*tcd/fyd
!
!
!Armadura mínima
!Voltando para MPa
fck=10*fck
fyd=10*fyd
!
!Romin da tração simples
!
a=2./3.
If (fck<=50) Then
    r1min=0.39*(fck**a)/fyd
Else
    r1min=2.756*Log(1+0.11*fck)/fyd

```



```

End If
!
!Romin da flexão simples
!
If(fck<=50) Then
  r2min=0.078*(fck**a)/fyd
Else
  r2min=0.5512*Log(1+0.11*fck)/fyd
End If
If (r2min<0.0015) Then
  r2min=0.0015
End If
!
If (idom.Eq.2) Then
  ! A solução do problema caiu no domínio 2 ou 3
  romin=r2min
Else
  ! A solução caiu no domínio 1
  ! A taxa mínima será interpolada entre r1min e r2min
  romin=r2min+(r1min-r2min)*(ami0-ami)/ami0
End If
!Armadura mínima
!
asmin=romin*b*h
!
If ((idom.Eq.2) .And. (aas<asmin)) Then
  ! Domínios 2 e 3: apenas aas respeita a armadura mínima
  aas=asmin
End If
!
If (idom.Eq.1) Then
  ! Domínio 1: a soma aas+asl respeita a armadura mínima
  astot=aas+asl
  If (astot<asmin) Then
    aas=aas*asmin/astot
    asl=asl*asmin/astot
  End If
End If
!
!
!MOSTRAR O RESULTADO
!
write(*,30)aas,asl
30 format(/,5x,'As (cm2)=' ,f6.2,4x,'Asl (cm2)=' ,f6.2)
!Encerrando o programa
100 a=0
End
!
SUBROUTINE Tensao(es,esl,fyd,tsl)
IMPLICIT REAL*8(A-H,O-Z)
!
!Calcula a tensão no aço
!es = módulo de elasticidade do aço em kN/cm2
!esl = deformação de entrada
!fyd = tensão de escoamento de cálculo em kN/cm2
!tsl = tensão de saída em kN/cm2
!

```

```

!Trabalhando com deformação positiva
ess=Abs(esl)
eyd=fyd/es
If(ess<eyd)Then
    tsl=es*ess
Else
    tsl=fyd
End If
!Trocando o sinal se necessário
If(esl<0)Then
    tsl=-tsl
End If
Return
End

```

9.4 – Programa em Visual Basic 2017

Na fig. 9.4.1 apresenta-se o formulário do programa escrito em Visual Basic 2017. Os dados podem ser fornecidos usando o ponto ou a vírgula como separador decimal.

Para criar o projeto, procede-se como descrito na seção 1.4 do capítulo 1. Utilizando a Caixa de Ferramentas, são introduzidos os controles no Form1. A propriedade Text do Form1 é “Flexo-Tração Normal Com Armaduras Assimétricas”, como aparece no topo do formulário. O arquivo com a listagem do código-fonte em Visual Basic 2017 pode ser obtido em www.editoradunas.com.br/programas.htm. Após a elaboração do formulário, basta clicar com o botão direito do mouse sobre o formulário para Exibir Código e colar o código-fonte.

The screenshot shows a Windows form titled "Flexo-Tração Normal Com Armaduras Assimétricas". On the left, there is a diagram of a rectangular cross-section of a beam. The width is labeled 'b', the total height is 'h', and the effective depth is 'd'. The top reinforcement area is labeled 'A's' and the bottom reinforcement area is labeled 'As'. On the right side of the form, there are several groups of input fields:

- Materiais:** fck (MPa)=, fyd (MPa)=, Es (GPa)=
- Coefficientes parciais:** Gamac =, Gamas =, Gamaf =
- Coefficiente de redistribuição:** Beta =
- Dimensões:** b (cm)=, h (cm)=, d (cm)=, d' (cm)=
- Esforços solicitantes de serviço:** Mk (kNm)=, Nk (kN)=

At the bottom, there are output fields for 'Resposta': As (cm2)= and A's (cm2)=, and a 'CALCULAR' button.

Fig. 9.4.1 – Formulário do programa em Visual Basic 2017

O formulário contém os seguintes controles:

Tipo	Nome	Descrição
PictureBox	PictureBox1	Imagem da seção
Label	Label1	fck (MPa)=
TextBox	TextBox1	Valor de fck
Label	Label2	fyk (MPa)=
TextBox	TextBox2	Valor de fyk
Label	Label3	Es (GPa)=
TextBox	TextBox3	Valor de Es
Label	Label4	Gamac =
TextBox	TextBox4	Valor de Gamac
Label	Label5	Gamas =
TextBox	TextBox5	Valor de Gamas
Label	Label6	Gamaf =
TextBox	TextBox6	Valor de Gamaf
Label	Label7	Beta =
TextBox	TextBox7	Valor de Beta
Label	Label8	b (cm)=
TextBox	TextBox8	Valor de b
Label	Label9	h (cm)=
TextBox	TextBox9	Valor de h
Label	Label10	d (cm)=
TextBox	TextBox10	Valor de d'
Label	Label11	d' (cm)=
TextBox	TextBox11	Valor de d'
Label	Label12	Mk (kNm)=
TextBox	TextBox12	Valor de Mk
Label	Label13	Nk (kN)=
TextBox	TextBox13	Valor de Nk
Label	Label14	As (cm ²)=
TextBox	TextBox14	Valor de As
Label	Label15	A's (cm ²)=
TextBox	TextBox15	Valor de A's
Button	Button1	CALCULAR
GroupBox	GroupBox1	Materiais
GroupBox	GroupBox2	Coeficientes parciais
GroupBox	GroupBox3	Coeficiente de redistribuição
GroupBox	GroupBox4	Dimensões
GroupBox	GroupBox5	Esforços solicitantes de serviço
GroupBox	GroupBox6	Resposta

LISTAGEM DO CÓDIGO-FONTE EM VISUAL BASIC 2017

```
Public Class Form1
    Public ts1 As Double
```

```
Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
    Dim fck, fyk, es, gamac, gamas, gamaf, bduct As Double
    Dim b, h, d, dl, amk, ank As Double
    Dim alamb, alfac, eu, qlim As Double
```

```

Dim fcd, tcd, fyd, amd, aand As Double
Dim delta, ani, ami, ami0 As Double
Dim idom As Integer
Dim wl, w, amisd, amilim, qsi, qsia As Double
Dim esl, aas, asl, a, r1min, r2min, romin, asmin, astot As Double
,
,
'Esta sub-rotina corresponde ao botão CALCULAR.
,
,
,
'Dimensionamento de seções retangulares à Flexo-Tração Normal com Armaduras Assimétricas
,
,
'ENTRADA DE DADOS
,
'Os dados são lidos das caixas de texto do formulário
,
'Resistência característica à compressão do concreto em MPa
'Substituindo a vírgula por ponto
TextBox1.Text = Replace(TextBox1.Text, ",", ".")
fck = Val(TextBox1.Text)
,
'Tensão de escoamento característica do aço em MPa
TextBox2.Text = Replace(TextBox2.Text, ",", ".")
fyk = Val(TextBox2.Text)
,
'Módulo de elasticidade do aço em GPa
TextBox3.Text = Replace(TextBox3.Text, ",", ".")
es = Val(TextBox3.Text)
,
'Coeficientes parciais de segurança:
'para o concreto
TextBox4.Text = Replace(TextBox4.Text, ",", ".")
gamac = Val(TextBox4.Text)
,
'para o aço
TextBox5.Text = Replace(TextBox5.Text, ",", ".")
gamas = Val(TextBox5.Text)
,
'para o momento fletor
TextBox6.Text = Replace(TextBox6.Text, ",", ".")
gamaf = Val(TextBox6.Text)
,
'Coeficiente beta de redistribuição de momentos
TextBox7.Text = Replace(TextBox7.Text, ",", ".")
bduct = Val(TextBox7.Text)
,
'Largura da seção transversal
TextBox8.Text = Replace(TextBox8.Text, ",", ".")
b = Val(TextBox8.Text)
,
'Altura da seção transversal
TextBox9.Text = Replace(TextBox9.Text, ",", ".")
h = Val(TextBox9.Text)
,
'Altura útil

```

```

TextBox10.Text = Replace(TextBox10.Text, ",", ".")
d = Val(TextBox10.Text)

```

```

'Parâmetro d' (distância da armadura A's até a
'borda da seção)

```

```

TextBox11.Text = Replace(TextBox11.Text, ",", ".")
dl = Val(TextBox11.Text)

```

```

'Momento fletor de serviço em kNm

```

```

TextBox12.Text = Replace(TextBox12.Text, ",", ".")
amk = Val(TextBox12.Text)

```

```

'Esforço normal de serviço em kN

```

```

TextBox13.Text = Replace(TextBox13.Text, ",", ".")
ank = Val(TextBox13.Text)

```

```

'FIM DA ENTRADA DE DADOS

```

```

'INÍCIO DOS CÁLCULOS

```

```

'Parâmetros do diagrama retangular

```

```

If (fck <= 50) Then

```

```

    alamb = 0.8

```

```

    alfac = 0.85

```

```

    eu = 3.5

```

```

    qlim = 0.8 * bduct - 0.35

```

```

Else

```

```

    alamb = 0.8 - (fck - 50) / 400

```

```

    alfac = 0.85 * (1 - (fck - 50) / 200)

```

```

    eu = 2.6 + 35 * ((90 - fck) / 100) ^ 4

```

```

    qlim = 0.8 * bduct - 0.45

```

```

End If

```

```

'Conversão de unidades: transformando para kN e cm

```

```

amk = 100 * amk

```

```

fck = fck / 10

```

```

fyk = fyk / 10

```

```

es = 100 * es

```

```

'Resistências de cálculo

```

```

fcd = fck / gamac

```

```

tcd = alfac * fcd

```

```

fyd = fyk / gamas

```

```

'Esforços solicitantes de cálculo

```

```

amd = gamaf * amk

```

```

aand = gamaf * ank

```

```

'Parâmetro geométrico

```

```

delta = dl / d

```

```

'Esforços solicitantes reduzidos

```

```

ani = aand / (b * d * tcd)

```

```

ami = amd / (b * d * d * tcd)

```

```

'Realização do dimensionamento

```

```

'Momento reduzido que define o final do domínio 1
,
ami0 = 0.5 * (1 - delta) * ani
,
If (ami <= ami0) Then
'O problema será resolvido no domínio 1
'Trata-se de flexo-tração com pequena excentricidade
'Variável para indicar o domínio
idom = 1
,
'Taxas mecânicas de armadura
wl = (ami0 - ami) / (1 - delta)
w = (ami0 + ami) / (1 - delta)
End If
,
If (ami > ami0) Then
'O problema será resolvido nos domínios 2 e 3
'Trata-se de flexo-tração com grande excentricidade
'Variável para indicar o domínio
idom = 2
,
'Momento equivalente
amisd = ami - ami0
'Momento limite
amilim = alamb * qlim * (1 - 0.5 * alamb * qlim)
,
If (amisd <= amilim) Then
'Armadura simples
qsi = (1 - Math.Sqrt(1 - 2 * amisd)) / alamb
w = alamb * qsi + ani
wl = 0
Else
'Armadura dupla
'Evitando armadura dupla no domínio 2
qsia = eu / (eu + 10)
If (qlim < qsia) Then
,
'Está resultando armadura dupla no domínio 2.
'Colocar mensagem para o usuário aumentar as dimensões da seção transversal e parar o
processamento
,
MsgBox("Resultou armadura dupla no domínio 2. Aumente as dimensões da seção
transversal", vbOKOnly + vbExclamation)
Exit Sub
End If
,
'Eliminando o caso em que qlim<delta
'Se isto ocorrer, a armadura de compressão estará tracionada
,
If (qlim <= delta) Then
,
'Colocar mensagem para o usuário aumentar as dimensões da seção transversal e parar o
processamento
,
MsgBox("Aumente as dimensões da seção transversal", vbOKOnly + vbExclamation)
Exit Sub
End If

```

```

    '
    'Deformação da armadura de compressão
    esl = eu * (qlim - delta) / qlim
    esl = esl / 1000
    'Tensão na armadura de compressão
    'Chamar sub-rotina
    Call Tensao(es, esl, fyd)
    '
    'Taxas mecânicas de armadura
    w1 = (amisd - amilim) * fyd / ((1 - delta) * tsl)
    w = alamb * qlim + (amisd - amilim) / (1 - delta) + ani
    End If
End If
'
'Áreas das armaduras
aas = w * b * d * tcd / fyd
asl = w1 * b * d * tcd / fyd
'
'
'Armadura mínima
'Voltando para MPa
fck = 10 * fck
fyd = 10 * fyd
'
'Romin da tração simples
'
a = 2 / 3
If (fck <= 50) Then
    r1min = 0.39 * (fck ^ a) / fyd
Else
    r1min = 2.756 * Math.Log(1 + 0.11 * fck) / fyd
End If
'
'Romin da flexão simples
'
If (fck <= 50) Then
    r2min = 0.078 * (fck ^ a) / fyd
Else
    r2min = 0.5512 * Math.Log(1 + 0.11 * fck) / fyd
End If
If (r2min < 0.0015) Then
    r2min = 0.0015
End If
'
If (idom = 2) Then
    'A solução do problema caiu no domínio 2 ou 3
    romin = r2min
Else
    'A solução caiu no domínio 1
    'A taxa mínima será interpolada entre r1min e r2min
    romin = r2min + (r1min - r2min) * (ami0 - ami) / ami0
End If
'Armadura mínima
'
asmin = romin * b * h
'
If ((idom = 2) And (aas < asmin)) Then

```

```

'Domínios 2 e 3: apenas aas respeita a armadura mínima
aas = asmin
End If
'
If (idom = 1) Then
'Domínio 1: a soma aas+asl respeita a armadura mínima
astot = aas + asl
If (astot < asmin) Then
aas = aas * asmin / astot
asl = asl * asmin / astot
End If
End If
'Convertendo a saída para duas casas decimais
aas = FormatNumber(aas, 2)
asl = FormatNumber(asl, 2)
'
'MOSTRAR O RESULTADO
'
TextBox14.Text = aas
TextBox15.Text = asl
'

End Sub
Private Sub Tensao(es, esl, fyd)
Dim ess, eyd As Double
'
'Calcula a tensão no aço
'es = módulo de elasticidade do aço em kN/cm2
'esl = deformação de entrada
'fyd = tensão de escoamento de cálculo em kN/cm2
'tsl = tensão de saída em kN/cm2
'
'Trabalhando com deformação positiva
ess = Math.Abs(esl)
eyd = fyd / es
If (ess < eyd) Then
tsl = es * ess
Else
tsl = fyd
End If
'Trocando o sinal se necessário
If (esl < 0) Then
tsl = -tsl
End If
End Sub
End Class

```

9.5 – Programa em Visual C# 2017

Na fig. 9.5.1 apresenta-se o formulário do programa escrito em Visual C# 2017. O formulário é inteiramente análogo ao do programa em Visual Basic 2017. Os dados podem ser fornecidos usando o ponto ou a vírgula como separador decimal. Os controles são os mesmos do programa em Visual Basic.

O arquivo com a listagem do código-fonte em Visual C# 2017 pode ser obtido em www.editoradunas.com.br/programas.htm. Após a elaboração do formulário, basta clicar com o botão direito do mouse sobre o formulário para Exibir Código e colar o código-fonte.

Fig. 9.5.1 – Formulário do programa em Visual C# 2017

LISTAGEM DO CÓDIGO-FONTE EM VISUAL C# 2017

```
using System;
using System.Globalization;
using System.Windows.Forms;

namespace DIMFTN
{
    public partial class Form1 : Form
    {
        //Declarando variáveis públicas
        public double es, esl, fyd, tsl;
        public double aas, asl, w, wl;
        public int idom;

        public Form1()
        {
            InitializeComponent();
        }

        private void button1_Click(object sender, EventArgs e)
        {
            double fck, fyk, gamac, gamas, gamaf, bduct;
```

```

double b, h, d, dl, amk, ank;
double alamb, alfac, eu, qlim;
double fcd, tcd, amd, aand;
double delta, ani, ami, ami0;
double amisd, amilim, qsi, qsia;
double a, r1min, r2min, romin, asmin, astot;

//
//
//Esta sub-rotina corresponde ao botão CALCULAR.
//
//
//Dimensionamento de seções retangulares à Flexo-Tração Normal com Armaduras
Assimétricas
//
//
//ENTRADA DE DADOS
//
//Os dados são lidos das caixas de texto do formulário
//
//Resistência característica à compressão do concreto em MPa
//Substituindo a vírgula por ponto
textBox1.Text = textBox1.Text.Replace(",", ".");
fck = Convert.ToDouble(textBox1.Text, CultureInfo.InvariantCulture);
//
//Tensão de escoamento característica do aço em MPa
textBox2.Text = textBox2.Text.Replace(",", ".");
fyk = Convert.ToDouble(textBox2.Text, CultureInfo.InvariantCulture);
//
//Módulo de elasticidade do aço em GPa
textBox3.Text = textBox3.Text.Replace(",", ".");
es = Convert.ToDouble(textBox3.Text, CultureInfo.InvariantCulture);
//
//Coeficientes parciais de segurança:
//para o concreto
textBox4.Text = textBox4.Text.Replace(",", ".");
gamac = Convert.ToDouble(textBox4.Text, CultureInfo.InvariantCulture);
//
//para o aço
textBox5.Text = textBox5.Text.Replace(",", ".");
gamas = Convert.ToDouble(textBox5.Text, CultureInfo.InvariantCulture);
//
//para o momento fletor
textBox6.Text = textBox6.Text.Replace(",", ".");
gamaf = Convert.ToDouble(textBox6.Text, CultureInfo.InvariantCulture);
//
//Coeficiente beta de redistribuição de
//momentos
textBox7.Text = textBox7.Text.Replace(",", ".");
bduct = Convert.ToDouble(textBox7.Text, CultureInfo.InvariantCulture);
//
//Largura da seção transversal
textBox8.Text = textBox8.Text.Replace(",", ".");
b = Convert.ToDouble(textBox8.Text, CultureInfo.InvariantCulture);
//
//Altura da seção transversal

```

```

textBox9.Text = textBox9.Text.Replace(",", ".");
h = Convert.ToDouble(textBox9.Text, CultureInfo.InvariantCulture);
//
//Altura útil
textBox10.Text = textBox10.Text.Replace(",", ".");
d = Convert.ToDouble(textBox10.Text, CultureInfo.InvariantCulture);
//
//Parâmetro d'
textBox11.Text = textBox11.Text.Replace(",", ".");
dl = Convert.ToDouble(textBox11.Text, CultureInfo.InvariantCulture);
//
//Momento fletor de serviço em kNm
textBox12.Text = textBox12.Text.Replace(",", ".");
amk = Convert.ToDouble(textBox12.Text, CultureInfo.InvariantCulture);
//
//Esforço normal de serviço em kN
textBox13.Text = textBox13.Text.Replace(",", ".");
ank = Convert.ToDouble(textBox13.Text, CultureInfo.InvariantCulture);
//
//
//FIM DA ENTRADA DE DADOS
//
//INÍCIO DOS CÁLCULOS
//
//
//Parâmetros do diagrama retangular
if (fck <= 50)
{
    alamb = 0.8;
    alfac = 0.85;
    eu = 3.5;
    qlim = 0.8 * bduct - 0.35;
}
else
{
    alamb = 0.8 - (fck - 50) / 400;
    alfac = 0.85 * (1 - (fck - 50) / 200);
    a = (90 - fck) / 100;
    eu = 2.6 + 35 * Math.Pow(a, 4);
    qlim = 0.8 * bduct - 0.45;
}
//
//Conversão de unidades: transformando para kN e cm
amk = 100 * amk;
fck = fck / 10;
fyk = fyk / 10;
es = 100 * es;
//
//Resistências de cálculo
fcd = fck / gamac;
tcd = alfac * fcd;
fyd = fyk / gamas;
//
//Esforços solicitantes de cálculo
amd = gamaf * amk;
aand = gamaf * ank;
//

```

```

//Parâmetro geométrico
delta = dl / d;
//
//Esforços solicitantes reduzidos
ani = aand / (b * d * tcd);
ami = amd / (b * d * d * tcd);
//
//Realização do dimensionamento
//
//Momento reduzido que define o final do domínio 1
//
ami0 = 0.5 * (1 - delta) * ani;
//
if(ami <= ami0)
{
    //O problema será resolvido no domínio 1
    //Trata-se de flexo-tração com pequena excentricidade
    //Variável para indicar o domínio
    idom = 1;
    //
    //Taxas mecânicas de armadura
    w1 = (ami0 - ami) / (1 - delta);
    w = (ami0 + ami) / (1 - delta);
}
//
if(ami > ami0)
{
    //O problema será resolvido nos domínios 2 e 3
    //Trata-se de flexo-tração com grande excentricidade
    //Variável para indicar o domínio
    idom = 2;
    //
    //Momento equivalente
    amisd = ami - ami0;
    //Momento limite
    amilim = alamb * qlim * (1 - 0.5 * alamb * qlim);
    //
    if(amisd <= amilim)
    {
        //Armadura simples
        qsi = (1 - Math.Sqrt(1 - 2 * amisd)) / alamb;
        w = alamb * qsi + ani;
        w1 = 0;
    }
    else
    {
        //Armadura dupla
        //Evitando armadura dupla no domínio 2
        qsia = eu / (eu + 10);
        if(qlim < qsia)
        {
            //
            //Está resultando armadura dupla no domínio 2.
            //Colocar mensagem para o usuário aumentar as dimensões da seção transversal e parar
            o processamento
            //

```

```

        MessageBox.Show("Resultou armadura dupla no domínio 2. Aumente as dimensões da
seção transversal");
        return;
    }
    //
    //Eliminando o caso em que qlim<delta
    //Se isto ocorrer, a armadura de compressão estará tracionada
    //
    if(qlim <= delta)
    {
        //
        //Colocar mensagem para o usuário aumentar as dimensões da seção transversal e parar
o processamento
        //
        MessageBox.Show("Aumente as dimensões da seção transversal");
        return;
    }
    //
    //Deformação da armadura de compressão
    esl = eu * (qlim - delta) / qlim;
    esl = esl / 1000;
    //Tensão na armadura de compressão
    //Chamar sub-rotina
    Tensao();
    //
    //Taxas mecânicas de armadura
    wl = (amisd - amilim) * fyd / ((1 - delta) * tsl);
    w = alamb * qlim + (amisd - amilim) / (1 - delta) + ani;
}
}
//
//Áreas das armaduras
aas = w * b * d * tcd / fyd;
asl = wl * b * d * tcd / fyd;
//
//
//Armadura mínima
//Voltando para MPa
fck = 10 * fck;
fyd = 10 * fyd;
//
//
a = 2.0 / 3.0;
if(fck <= 50)
{
    r1min = 0.39 * Math.Pow(fck, a) / fyd;
}
else
{
    r1min = 2.756 * Math.Log(1 + 0.11 * fck) / fyd;
}
//
//Romin da flexão simples
//
if(fck <= 50)
{
    r2min = 0.078 * Math.Pow(fck, a) / fyd;

```

```

    }
else
{
    r2min = 0.5512 * Math.Log(1 + 0.11 * fck) / fyd;
}
if(r2min < 0.0015)
{
    r2min = 0.0015;
}
//
if(idom == 2)
{
    //A solução do problema caiu no domínio 2 ou 3
    romin = r2min;
}
else
{
    //A solução caiu no domínio 1
    //A taxa mínima será interpolada entre r1min e r2min
    romin = r2min + (r1min - r2min) * (ami0 - ami) / ami0;
}
//Armadura mínima
//
asmin = romin * b * h;
//
if(idom == 2)
{
    if (aas < asmin)
    {
        //Domínios 2 e 3: apenas aas respeita a armadura mínima
        aas = asmin;
    }
}
if(idom == 1)
{
    //Domínio 1: a soma aas+asl respeita a armadura mínima
    astot = aas + asl;
    if(astot < asmin)
    {
        aas = aas * asmin / astot;
        asl = asl * asmin / astot;
    }
}
//Convertendo a saída para duas casas decimais
//
decimal saida1 = Decimal.Round(Convert.ToDecimal(aas), 2);
decimal saida2 = Decimal.Round(Convert.ToDecimal(asl), 2);
//
//MOSTRAR O RESULTADO
//
textBox14.Text = Convert.ToString(saida1);
textBox15.Text = Convert.ToString(saida2);
//
}
private void Tensao()
{
    double ess, eyd;

```

```

//
//Calcula a tensão no aço
//es = módulo de elasticidade do aço em kN/cm2
//esl = deformação de entrada
//fyd = tensão de escoamento de cálculo em kN/cm2
//tsl = tensão de saída em kN/cm2
//
//Trabalhando com deformação positiva
ess = Math.Abs(esl);
eyd = fyd / es;
if (ess < eyd)
{
    tsl = es * ess;
}
else
{
    tsl = fyd;
}
//Trocando o sinal se necessário
if (esl < 0)
{
    tsl = -tsl;
}
}
}

```

9.6 – Exemplos para testar os programas

Os resultados apresentados a seguir servem para o leitor testar os programas. O dimensionamento à flexo-tração normal é feito para a seção da fig. 9.6.1. Esses exemplos foram resolvidos manualmente no Capítulo 2 do Volume 4.

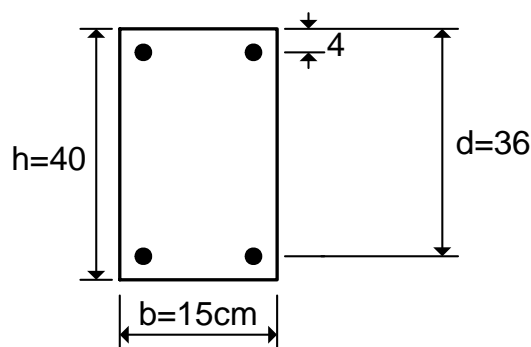


Fig. 9.6.1 - Seção transversal

São dados do problema:

$$f_{ck} = 20 \text{ MPa}; \quad f_{yk} = 500 \text{ MPa}; \quad E_s = 200 \text{ GPa}$$

$$\gamma_c = 1,4; \quad \gamma_s = 1,15; \quad \gamma_f = 1,4; \quad \beta = 1$$

O esforço normal é $N_k = 214,29 \text{ kN}$ para todos os exemplos.

Na tabela 9.6.1, apresentam-se os resultados obtidos.

Tabela 9.6.1 - Resultados do dimensionamento

f_{ck} (MPa) →		20		50		70	
Ex.	M_k (kNm)	A_s (cm ²)	A'_s (cm ²)	A_s (cm ²)	A'_s (cm ²)	A_s (cm ²)	A'_s (cm ²)
1	28,57	6,32	0,58	6,32	0,58	6,32	0,58
2	57,14	9,11	0	9,00	0	8,99	0
3	114,29	15,37	3,04	14,91	0	14,70	0

Editora Dunas
Rua Tiradentes, 105 – Bairro Cidade Nova
96211-080 Rio Grande – RS
www.editoradunas.com.br
e-mail: contato@editoradunas.com.br