

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

JOSÉ MILTON DE ARAÚJO

Professor Titular – Escola de Engenharia da FURG
Doutor em Engenharia

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

DUNAS

PROGRAMAS PARA DIMENSIONAMENTO E VERIFICAÇÃO DE CONCRETO ARMADO

© Copyright Editora DUNAS

A663c Araújo, José Milton de
Programas para dimensionamento e verificação de concreto armado /
José Milton de Araújo. - Rio Grande: Dunas, 2018. 1.ed.

Bibliografia

1. Concreto armado. I. Título

CDU 624.012.45

CDD 624.1834

ISBN 978-85-86717-19-2

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

APRESENTAÇÃO

Este livro é um complemento da obra Curso de Concreto Armado. 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.

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.

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. Para estudantes e professores, o Fortran pode ser obtido gratuitamente no site da Intel². Ele faz parte do Intel® Parallel Studio XE for Windows 2018. 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.

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 separado 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 texto 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.

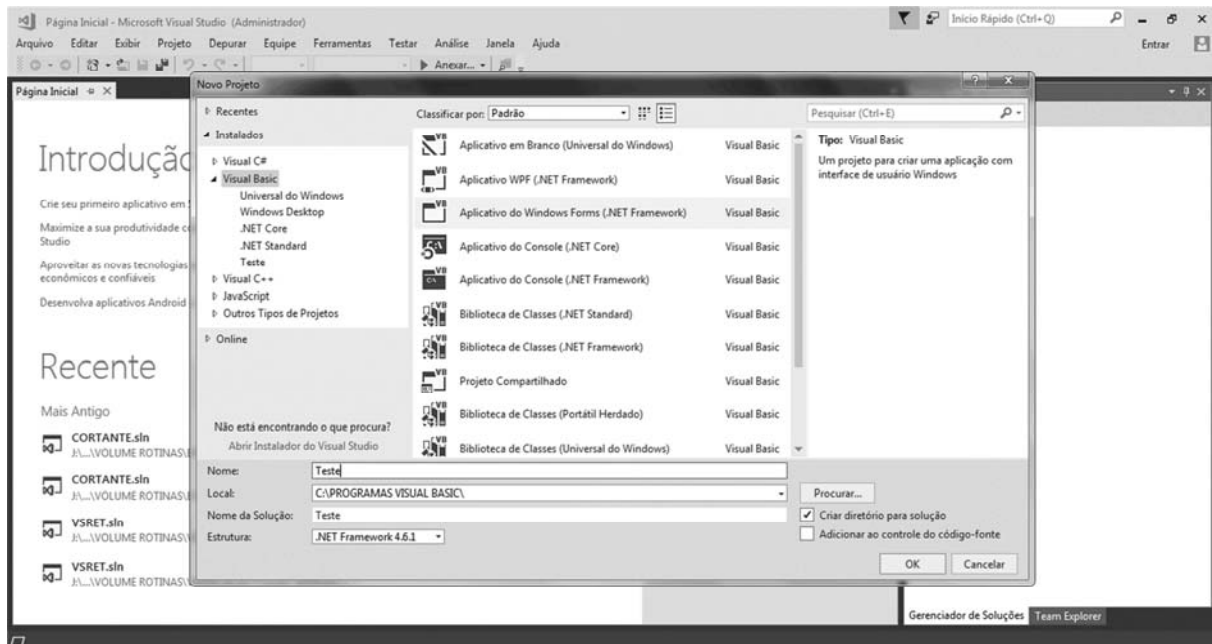


Fig. 1.4.1 – Iniciando novo projeto no Visual Studio 2017

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.

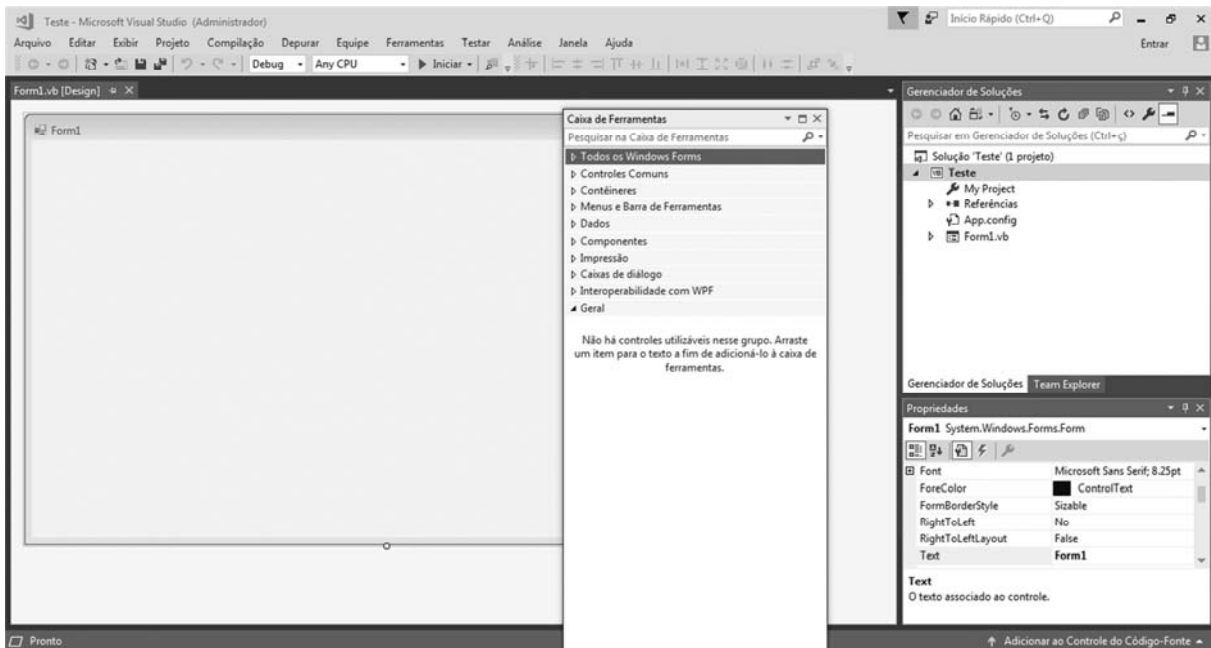


Fig. 1.4.2 – Janela com o Form1 e a Caixa de Ferramentas

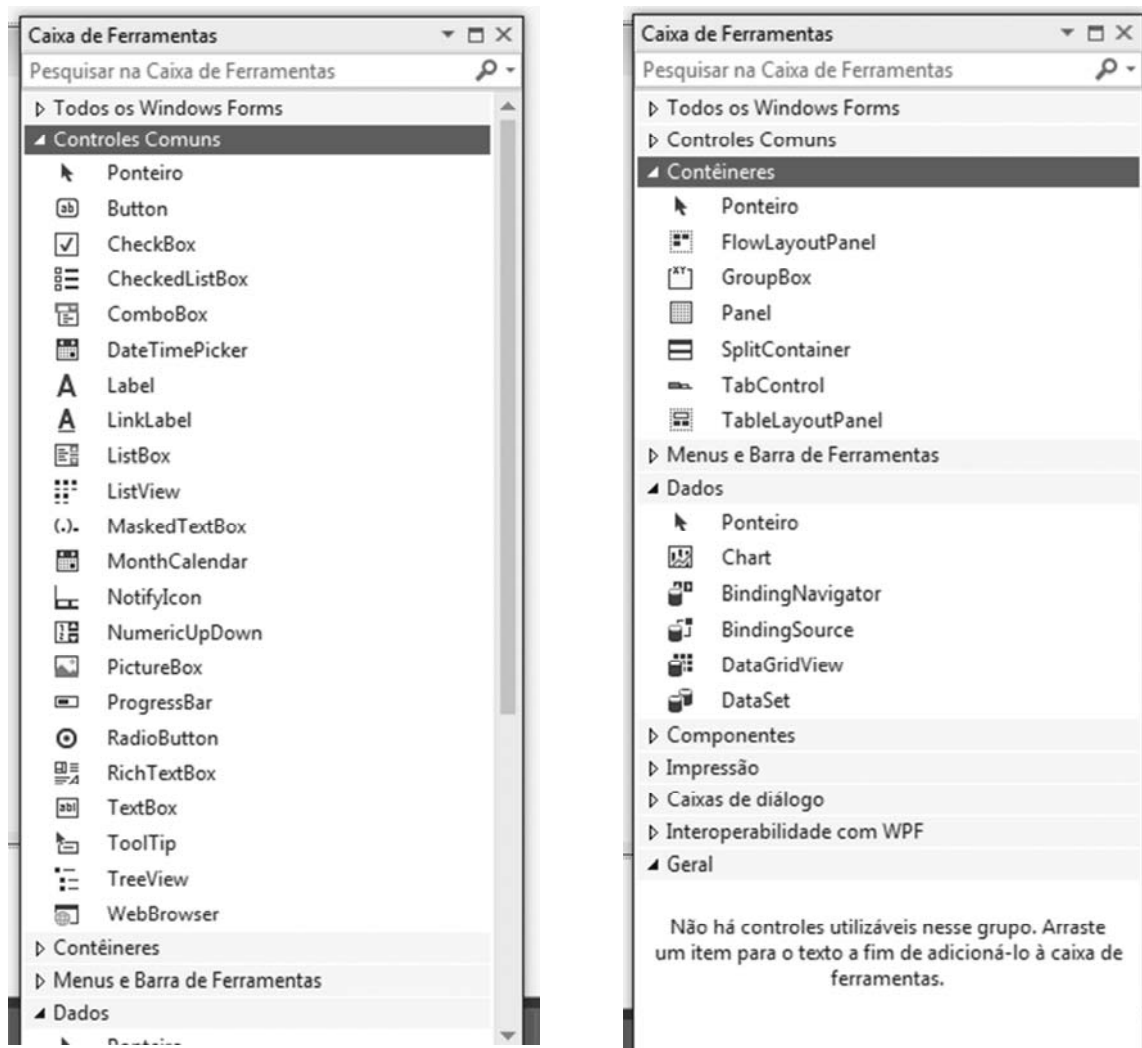


Fig. 1.4.3 – Controles da Caixa de Ferramentas

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 acrescentadas 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

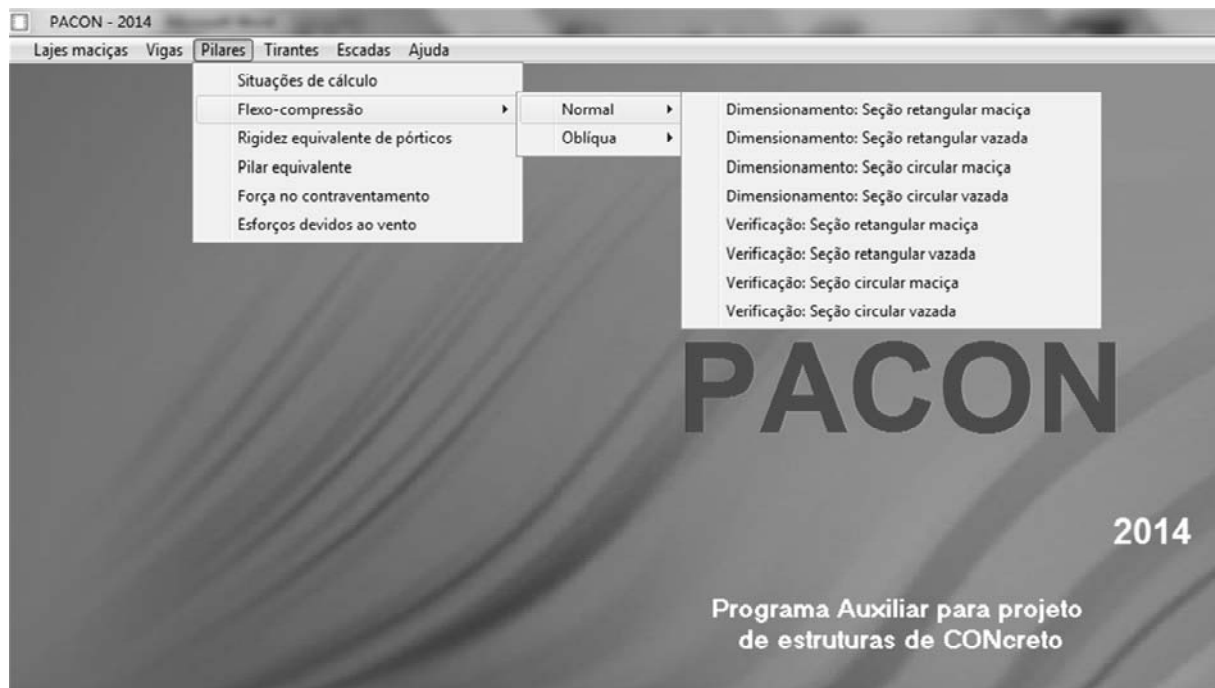


Fig. 1.6.3 - Módulo Pilares do PACON

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.

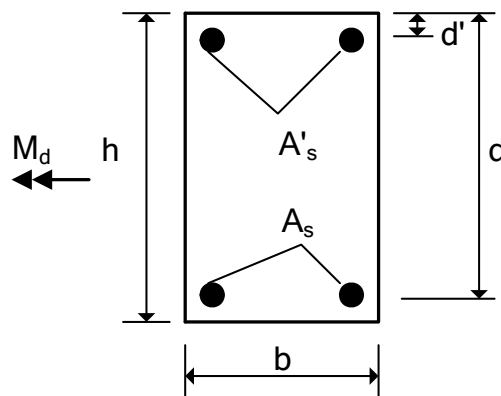


Fig. 2.1.1 - Seção retangular submetida à flexão simples normal

Os dados do problema são os seguintes:

- propriedades dos materiais: f_{ck} , f_{yk} , E_s ;
- coeficientes parciais de segurança: γ_c , γ_s , γ_f
- coeficiente de redistribuição de momentos: $0,75 \leq \beta \leq 1$
- dimensões da seção transversal: b , h , d , d' ;
- 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:

$$\lambda = 0,8$$

$$\alpha_c = 0,85$$

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

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

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

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

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

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

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

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

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

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:

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

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:

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

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  
!  
!Declarnado 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
```

```

!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 =

LISTAGEM DO CÓDIGO-FONTE EM VISUAL BASIC 2017

Public Class Form1

Public Inherited 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)

,

,

,

,

,

,

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(",", ".");
```



```

    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.

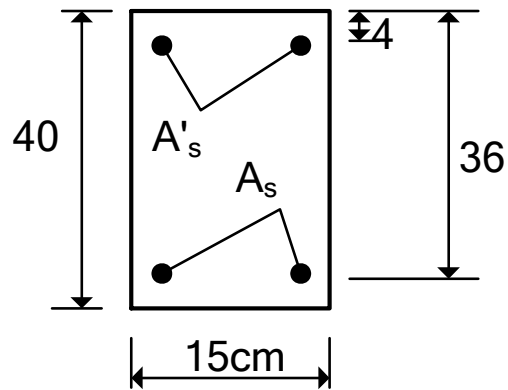


Fig. 2.6.1 - Dimensões da seção retangular

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.

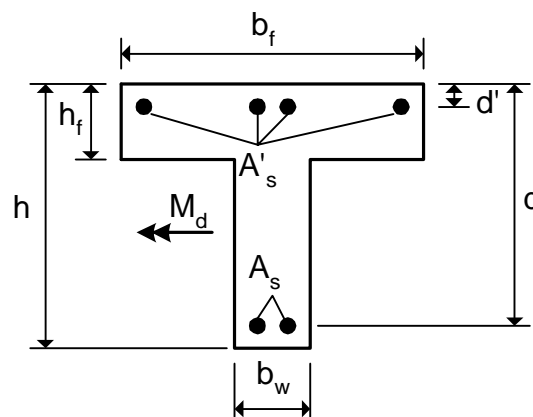


Fig. 3.1.1 - Seção T submetida à flexão simples normal

Os dados do problema são os seguintes:

- propriedades dos materiais: f_{ck} , f_{yk} , E_s
- coeficientes parciais de segurança: γ_c , γ_s , γ_f
- coeficiente de redistribuição de momentos: $0,75 \leq \beta \leq 1$
- dimensões da seção transversal: b_f , h_f , b_w , h , d , d'
- 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:

$$\lambda = 0,8$$

$$\alpha_c = 0,85$$

$$\varepsilon_u (\text{o/oo}) = 3,5$$

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

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

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

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

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

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

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

$$\beta_f = h_f/d$$

$$\beta_w = b_w/b_f$$

B) Sub-rotina tensãoParâmetros 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 tensãoSe $\varepsilon_s < 0$, 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
```

```

    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.

Fig. 3.4.1 – Formulário do programa em Visual Basic 2017

O formulário contém os seguintes controlos:

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


```
'  
'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
```

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.
        }
    }
}

```

```

//
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;
}
}
}
}

```

3.6 – Exemplos para testar os programas

Os resultados apresentados a seguir servem para o leitor testar os programas. Os exemplos correspondem à seção T da fig. 3.6.1. Esses exemplos foram resolvidos manualmente no Capítulo 5 do Volume 1.

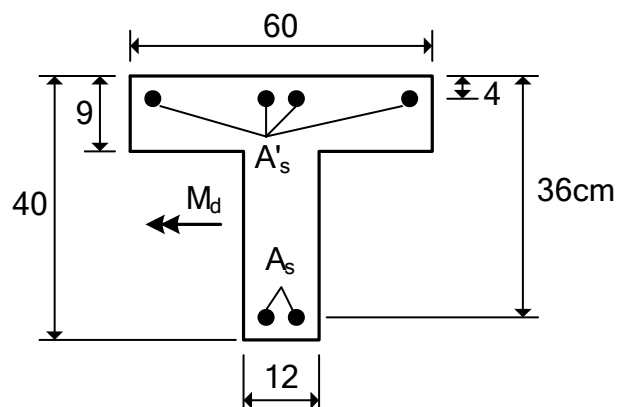


Fig. 3.6.1 - Dimensões da seção T

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.

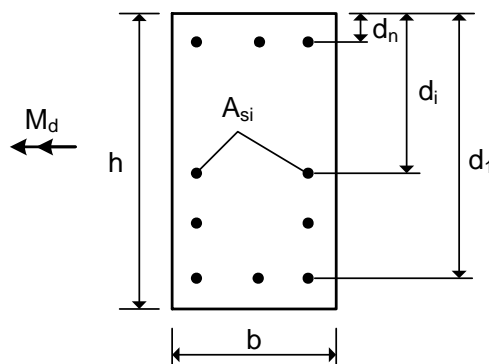


Fig. 4.1.1 - Seção retangular com várias camadas de armadura

Os dados do problema são os seguintes:

- propriedades dos materiais: f_{ck} , f_{yk} , E_s
- coeficientes parciais de segurança: γ_c , γ_s
- largura da seção transversal: b
- número de camadas de armadura: n
- 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.

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.

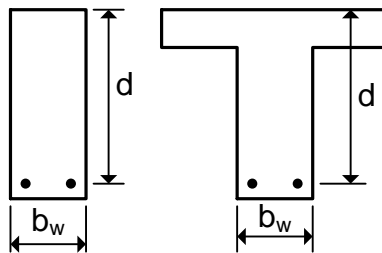


Fig. 5.1.1 - Seção retangular e seção T

Os dados do problema são os seguintes:

- propriedades dos materiais: f_{ck} , f_{yk}
- coeficientes parciais de segurança: γ_c , γ_s , γ_f
- largura da nervura seção transversal: b_w
- altura útil: d
- 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.

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.

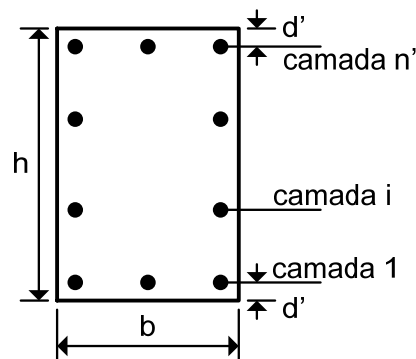


Fig. 6.1.1 - Seção retangular com várias camadas de armadura

Os dados do problema são os seguintes:

- propriedades dos materiais: f_{ck} , f_{yk} , E_s
- coeficientes parciais de segurança: γ_c , γ_s , γ_f
- dimensões da seção transversal: b , h , d'
- número de camadas de armadura: $n' \geq 2$
- número de barras de cada camada: ≥ 2
- 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: $n_i(i) \geq 2$

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.

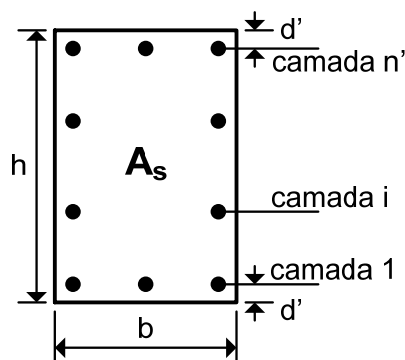


Fig. 7.1.1 - Seção retangular com várias camadas de armadura

Os dados do problema são os seguintes:

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

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).

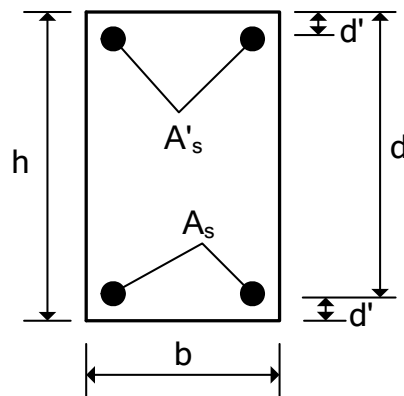


Fig. 8.1.1 - Seção transversal

Os dados do problema são os seguintes:

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

Capítulo 9

DIMENSIONAMENTO À FLEZO-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.

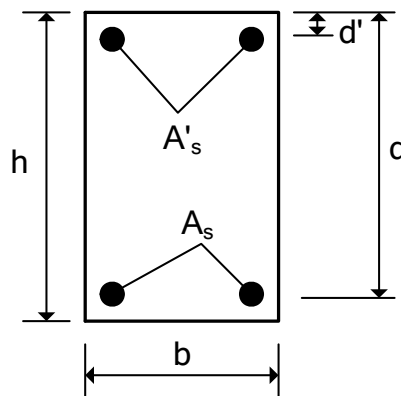


Fig. 9.1.1 - Seção transversal com armaduras assimétricas

Os dados do problema são os seguintes:

- propriedades dos materiais: f_{ck} , f_{yk} , E_s
- coeficientes parciais de segurança: γ_c , γ_s , γ_f
- coeficiente de redistribuição de momentos: $0,75 \leq \beta \leq 1$
- dimensões da seção transversal: b , h , d , d'
- momento fletor de serviço: M_k
- 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