



■ ■ série livros didáticos informática ufrgs ■ ■

algoritmos e programação

com exemplos em Pascal e C

■ ■ nina edelweiss

■ ■ maria aparecida castro livi



...→ as autoras

Nina Edelweiss é engenheira eletricista e doutora em Ciência da Computação pela Universidade Federal do Rio Grande do Sul. Durante muitos anos, lecionou em cursos de Engenharia e de Ciência da Computação na UFRGS, na UFSC e na PUCRS. Foi, ainda, orientadora do Programa de Pós-Graduação em Ciência da Computação da UFRGS. É coautora de três livros, tendo publicado diversos artigos em periódicos e em anais de congressos nacionais e internacionais. Participou de diversos projetos de pesquisa financiados por agências de fomento como CNPq e FAPERGS, desenvolvendo pesquisas nas áreas de bancos de dados e desenvolvimento de software.

Maria Aparecida Castro Livi é licenciada e bacharel em Letras, e mestre em Ciência da Computação pela Universidade Federal do Rio Grande do Sul. Desenvolveu sua carreira profissional na UFRGS, onde foi programadora e analista de sistema, antes de ingressar na carreira docente. Ministrou por vários anos a disciplina de Algoritmos e Programação para alunos dos cursos de Engenharia da Computação e Ciência da Computação. Sua área de interesse prioritário é o ensino de Linguagens de Programação, tanto de forma presencial quanto a distância.



E22a Edelweiss, Nina.
Algoritmos e programação com exemplos em Pascal e C
[recurso eletrônico] / Nina Edelweiss, Maria Aparecida Castro
Livi. – Dados eletrônicos. – Porto Alegre : Bookman, 2014.

Editado também como livro impresso em 2014.
ISBN 978-85-8260-190-7

1. Informática. 2. Algoritmos – Programação. I. Livi,
Maria Aparecida Castro. II. Título.

CDU 004.421

Catálogo na publicação: Ana Paula M. Magnus – CRB 10/2052



capítulo

5

estruturas de repetição

■ ■ Este capítulo apresenta comandos de repetição utilizados para implementar iterações de conjuntos de comandos. Após a introdução dos conceitos de laço de repetição e de contador, são apresentados e discutidos os comandos de repetição por contagem, de repetição condicional por avaliação prévia de uma condição e de repetição condicional por avaliação posterior de uma condição.

Nos algoritmos vistos nos capítulos anteriores, os comandos são executados em sequência, um após o outro, uma única vez ou até mesmo nenhuma vez, dependendo de uma condição especificada. Este capítulo introduz uma nova situação, bastante comum na programação: sequências de comandos que são executados repetidas vezes. O número de repetições pode ser conhecido *a priori* ou pode estar associado à ocorrência de uma condição que se verifique ao longo do processamento.

No capítulo anterior, foi resolvido o problema do cálculo da média aritmética de três notas de um aluno, com a determinação de sua aprovação ou reprovação, complementado ainda pelo cálculo do conceito correspondente à média obtida. Com frequência, esse processo é repetido para um número determinado de alunos de uma turma. Nesse caso, os dados de cada aluno são obtidos, processados e informados de forma semelhante, porém de forma independente, sendo esse processo repetido tantas vezes quantos forem os alunos da turma – um número conhecido, utilizado como limite no controle das repetições.

Existem também situações em que o número de repetições de um conjunto de comandos não pode ser estabelecido previamente. No exemplo de cálculo dos conceitos, mesmo conhecendo-se o número total de alunos da turma pode acontecer de não se ter as notas de todos eles, porque um ou mais deixou de fazer as provas ou desistiu do curso. Quando não se deseja processar os dados de todos os alunos, o final da entrada de dados pode ser limitado através, por exemplo, do fornecimento de um conjunto de notas com valores nulos, ou do código correspondente ao último aluno a ser analisado. Portanto, nos casos em que o número de repetições não é previamente conhecido, o encerramento das repetições é controlado por uma condição que é verificada ao longo do processamento.

As sequências de comandos que são repetidas duas ou mais vezes são também chamadas de **laços de repetição**. Para criar um laço de repetição, além da ação ou ações que nele devam ser executadas, devem ser definidos:

1. uma forma para indicar o retorno a um ponto determinado do código, para repetir o laço (o que, até este momento, não esteve disponível);
2. um recurso para registrar o número de vezes que o laço foi realizado, o que remete para o uso de um contador a ser alterado cada vez que o laço for executado;
3. uma condição que, testada ao final da execução do laço, permita determinar se ele deve ser repetido novamente ou não.

Os elementos acima são, em sua maioria, fornecidos nos comandos iterativos que permitem implementar repetições. São eles: comando de repetição por contagem, comando de repetição condicional por avaliação prévia de condição e comando de repetição condicional por avaliação posterior de condição.

5.1

→ conceito de contador

O conceito de uma variável que atua como um **contador** está relacionado a repetições por contagem. Esse é o caso, por exemplo, de uma roleta colocada na entrada de um centro de

eventos com a finalidade de contar quantas pessoas entram no local. A roleta, inicialmente zerada, soma uma unidade a cada pessoa que por ela passa. Essa informação permite inclusive que a entrada de clientes seja encerrada quando o número registrado pela roleta atingir a lotação máxima.

Dessa forma, uma variável do tipo contador pode ser utilizada tanto para contabilizar o número de ocorrências de determinada ação como para determinar o seu encerramento. O contador deve ser sempre:

1. inicializado, normalmente com o valor zero, antes de iniciar a contagem:
`contador ← 0`
2. incrementado, sempre que uma nova ocorrência do que está sendo contado for identificada ou processada. A forma de incrementar um contador é atribuindo a ele seu valor atual, incrementado de uma unidade:
`contador ← contador + 1`

5.2

→ comando de repetição por contagem **para/faça**

O **comando de repetição por contagem para/faça** faz que a execução de uma ação, ou grupo de ações, seja repetida um número predefinido de vezes. Isso é feito vinculando a execução de um ou mais comandos ao valor de uma variável de controle, com funcionamento análogo ao de um contador. O controle das repetições é definido através de um cabeçalho, no qual são definidos o nome da variável de controle, seus valores inicial e final, e o valor do incremento que a variável de controle deve receber após cada repetição.

A sintaxe do comando repetição por contagem **para/faça** é:

```
para <variável de controle> de <valor inicial> [ incr <valor do incremento> ] até <valor final> faça  
  <comando>
```

Apenas variáveis ordinais simples podem ser utilizadas como variáveis de controle. Nesse tipo de variável, os valores válidos integram um conjunto ordenado de valores discretos, ou seja, se forem considerados três valores em sequência, entre o primeiro e o terceiro valor existirá tão somente um valor. Variáveis inteiras e tipo caractere são exemplos de variáveis ordinais simples.

Os valores inicial e final devem ser do mesmo tipo da variável de controle. Por exemplo, no cabeçalho:

```
para i de 1 incr 1 até 10 faça
```

a variável de controle `i` deve ser inteira. Já no cabeçalho:

```
para letra de 'a' incr 1 até 'm' faça
```

a variável de controle `letra` deve ser do tipo caractere.

Encerra aqui o trecho do livro disponibilizado para esta Unidade de Aprendizagem. Na Biblioteca Virtual da Instituição, você encontra a obra na íntegra.