

Aula prática 6

Comandos de repetição — while

Resumo

Nesta aula vamos trabalhar com problemas cuja solução envolve realizar um cálculo ou tarefa repetidas vezes, enquanto uma determinada condição é satisfeita. Em outras palavras, a implementação da solução de tais problemas requer o uso de um comando de repetição, tal como o comando **while**.

Sumário

1	Comandos de repetição - while	1
2	Exemplos	1
2.1	Exibindo uma sequência de números naturais	1
3	Resolvendo problemas	3

1 Comandos de repetição - while

A solução de diversos problemas, em computação, envolve a *repetição* de uma sequência de tarefas, ou comandos, enquanto uma determinada *condição* é satisfeita. Esse processo de repetição, ou *loop*, é implementado por meio do comando **while**, que tem a seguinte sintaxe:

```
while condição do  
    bloco de comandos while  
end
```

A *condição* deve ser uma expressão *booleana*, isto é, cujo valor é verdadeiro (%t) ou é falso (%f). O bloco de comandos é qualquer sequência de comandos, incluindo, possivelmente, comandos de atribuição, de entrada e saída, de desvio ou outros comandos de repetição.

A execução de um comando **while** é feita do seguinte modo:

1. a *condição* do **while** é avaliada;
2. se a condição avalia como %t (verdadeira), o *bloco de comandos while* é executado, e volta-se ao passo 1;
3. caso contrário, isto é, se a *condição* avalia para %f (falso), comando **while** termina. A execução do programa prossegue a partir do comando imediatamente subsequente ao **end** do comando **while**.

Note que, se a condição for inicialmente falsa, isto é, se o resultado for %f na primeira vez em que a condição é avaliada, o bloco de comandos **while** não é executado nenhuma vez. Por outro lado, se o valor da condição permanece sempre verdadeiro, em cada iteração do comando **while**, a execução desse comando prossegue indefinidamente.

2 Exemplos

2.1 Exibindo uma sequência de números naturais

Escrever uma aplicação para exibir a sequência dos números naturais menores que 25, usando a estrutura de repetição **while**.

Exemplo de execução da aplicação

Contagem dos número naturais até 25:

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25

O comando para exibir o número deverá ser escrito uma única vez no programa, dentro de um comando de repetição, e ele será repetido várias vezes. Em cada repetição ele exibirá um número diferente. Isto só será possível se usarmos uma variável para armazenar o número a ser exibido, e a cada repetição o valor da variável for modificado.

Assim para resolvermos este problema será usada uma variável para fazer a contagem dos números naturais de 0 até 25. Chamaremos esta variável de **contador**. O valor desta variável será exibido dentro do comando de repetição.

- O valor inicial da variável deverá ser 1, o primeiro valor a ser exibido.
- A cada repetição o valor da variável deverá ser incrementado em uma unidade, ou seja, **contador** receberá um novo valor igual ao seu sucessor do seu valor atual.

O corpo do comando de repetição deverá ser executado enquanto a variável **contador** estiver dentro do intervalo desejado. Logo a teste a ser usado no comando **while** deve verificar se **contador** é menor ou igual a 25.

Com estas considerações, chegamos ao seguinte programa:

```
clear;
clc;
printf("Contagem dos número naturais até 25\n");
contador = 1;
while contador <= 25 do
    printf("%g ", contador);
    contador = contador + 1;
end
```

3 Resolvendo problemas

Tarefa 1: Contagem dos números pares

Escreva um programa para exibir os números naturais pares dentro de um intervalo. O usuário deverá informar os limites do intervalo.

Não é necessário fazer a validação da entrada: assumiremos que o usuário é inteligente o suficiente para digitar valores corretos.

Exemplo de execução da aplicação

```
ontagem dos número naturais pares
=====
limite inferior do intervalo: 10
limite superior do intervalo: 25
10 12 14 16 18 20 22 24
=====
```

Solução:

```
clear;
clc;
printf("Contagem dos número naturais pares\n");
printf("=====\n");
a = input("limite inferior do intervalo: ");
b = input("limite superior do intervalo: ");
contador = a;
while contador <= b do
    printf("%g ", contador);
    contador = contador + 2;
end
printf("\n=====\n");
```

Tarefa 2: Cálculo da média aritmética

Escreva um programa para calcular a média aritmética das notas obtidas pelos alunos de *Programação de Computadores*. O usuário deverá primeiramente informar a quantidade de alunos da turma, e em seguida a nota obtida por cada aluno. Então o programa deve calcular e exibir a média aritmética das notas, caso o número de alunos seja positivo.

Exemplo de execução da aplicação

Cálculo da média aritmética das notas

=====

Informe a quantidade de alunos na turma: 0

Não há alunos na turma

Exemplo de execução da aplicação

Cálculo da média aritmética das notas

=====

Informe a quantidade de alunos na turma: 5

Informe a nota do aluno: 9.4

Informe a nota do aluno: 5

Informe a nota do aluno: 6.5

Informe a nota do aluno: 7.5

Informe a nota do aluno: 2.6

Média calculada: 6.20

Dicas:

- Use uma variável para armazenar o número total de alunos da turma.
- Use uma variável para calcular a soma das notas dos alunos. O seu valor inicial deve ser 0 (o elemento neutro da adição).
- Use uma variável para fazer a contagem dos alunos. O seu valor inicial deverá ser 1. Esta variável será usada para controlar as repetições.
- No corpo do comando de repetição:
 - leia a nota de um aluno, armazenando-a em uma variável,
 - atualize a variável da soma para acrescentar a nota lida à soma já calculada, e
 - atualize a variável contadora, para indicar que mais uma nota foi processada.
- Após a finalização da entrada das notas, verifique se a quantidade de alunos realmente é positiva. Neste caso calcule e exiba a média. Caso contrário exiba uma mensagem adequada.

Solução:

```

clear;
clc;
printf("Cálculo da média aritmética das notas\n");
printf("=====\n");
printf("\n");
quant = input("Informe a quantidade de alunos na turma: ")
printf("\n");
soma = 0;
contador = 1;
while contador <= quant do
    nota = input("Informe a nota do aluno: ");
    soma = soma + nota;
    contador = contador + 1;
end
printf("\n");
if quant > 0 then
    media = soma/quant;
    printf("Média calculada: %.2f\n", media)
else
    printf("Não há alunos na turma\n")
end

```

Tarefa 3: Portaria da festa Baranga 2014

No ginásio da UFOP ocorrerá a festa Baranga 2014/2. O ingresso masculino será de R\$ 12,50 e o feminino será de R\$ 7,40. Um calouro ficou encarregado de operar um programa na portaria do ginásio para controlar o acesso das pessoas à festa. O programa é executado da seguinte forma:

1. Quando chega um homem na festa, ele digita h.
2. Quando chega uma mulher na festa ele digita m.
3. Quando o calouro quiser encerrar a entrada de dados ele digita q.

O calouro não tem noção de quantas pessoas irão à festa.

No momento que a entrada de dados for encerrada, o programa calcula quanto foi arrecadado com os ingressos masculinos e com os ingressos femininos. Também é calculado o total arrecadado.

Codifique o programa operado pelo calouro na linguagem Scilab.

Exemplo de execução da aplicação

Portaria da festa BARANGA 2014/2

Quem chegou (h/m/q): w

Quem chegou (h/m/q): i

Quem chegou (h/m/q): p

Quem chegou (h/m/q): q

Quantidade de homens: 0

Quantidade de mulheres: 0

Arrecadação com homens: R\$ 0.00

Arrecadação com mulheres ...: R\$ 0.00

Arrecadação total: R\$ 0.00

Exemplo de execução da aplicação

Portaria da festa BARANGA 2014/2

Quem chegou (h/m/q): m

Quem chegou (h/m/q): m

Quem chegou (h/m/q): h

Quem chegou (h/m/q): m

Quem chegou (h/m/q): h

Quem chegou (h/m/q): m

Quem chegou (h/m/q): h

Quem chegou (h/m/q): m

Quem chegou (h/m/q): m

Quem chegou (h/m/q): h

Quem chegou (h/m/q): q

Quantidade de homens: 4

Quantidade de mulheres: 6

Arrecadação com homens: R\$ 50.00

Arrecadação com mulheres ...: R\$ 44.40

Arrecadação total: R\$ 94.40

Dicas:

- A entrada de dados deve ser textual. Logo lembre-se de usar um segundo argumento na chamada da função `input` para indicar que será lida uma string.
- Utilize uma variável para contar os homens, e outra para contar as mulheres.

- Incremente a variável adequada quando o usuário digitar h ou m.
- O comando de repetição deve encerrar quando o usuário digitar q.
- A primeira entrada de dados deve ser feita antes de começar a repetir, pois a condição de repetição depende do valor digitado pelo usuário.
- As demais entradas de dados devem ser feitas no corpo do comando de repetição.

Solução:

```
clear; clc;
printf("Portaria da festa BARANGA 2014/2\n");
printf("-----\n");

homens = 0;    // contador do número de homens
mulheres = 0; // contador do número de mulheres
quem = input("Quem chegou (h/m/q): ", "s");
while quem <> 'q'
    if quem == 'h' then
        homens = homens + 1;
    elseif quem == 'm' then
        mulheres = mulheres + 1;
    end
    quem = input("Quem chegou (h/m/q): ", "s");
end
arrecadacaoH = homens * 12.50;
arrecadacaoM = mulheres * 7.40;
arrecadacaoT = arrecadacaoH + arrecadacaoM;
printf("\nQuantidade de homens .....: %g", homens);
printf("\nQuantidade de mulheres .....: %g", mulheres);
printf("\nArrecadação com homens .....: R$ %.2f", arrecadacaoH);
printf("\nArrecadação com mulheres ...: R$ %.2f", arrecadacaoM);
printf("\nArrecadação total .....: R$ %.2f", arrecadacaoT);
```