INTRODUÇÃO

*Algoritmia - JavaScript –* Ciclos de repetição

As estruturas de repetição ou ciclos permitem repetir um conjunto de instruções de um programa, controlado pelo resultado lógico da avaliação de uma condição ou expressão.

# Ciclo *For* (Para)

O ciclo **for** é um tipo de estrutura de repetição em que inicialmente é conhecido o número de vezes que o bloco de instruções será repetido. Para tal, utiliza um contador automático implementado através de uma variável do tipo inteiro. O valor desta variável vai ser incrementado (***to***) ou decrementado (***downto***), resultando a realização de uma contagem crescente ou decrescente do número de repetições indicadas na estrutura do ciclo.

**Algoritmo med\_num;**

Início

Escrever (“Indique quantos inteiros quer digitar: “);

Ler (n);

soma ← 0

Para i ← 1 até n Fazer

Escrever (“Indique um número inteiro: “)

Ler (num);

soma ← soma + num;

FimPara

Media ← soma / n;

Escrever (“A média dos números é”, media);

Fim

1. Cria o fluxograma correspondente.
2. Apresenta a traçagem do algoritmo considerando os seguintes valores de entrada:

**n** = 2

**num** = 10 e 12

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | n | Soma←0 | i←0 | I <=n | num | soma←soma+n | i←i+1 | media←soma/n | Saída |
| 1º Passo | 2 |  |  |  |  |  |  |  |  |
| 2º Passo | 2 | 0 |  |  |  |  |  |  |  |
| 3º Passo | 2 | 0 | 1 |  |  |  |  |  |  |
| 4º Passo | 2 | 0 | 1 | 1 <= 2 |  |  |  |  |  |
| 5º Passo | 2 | 0 | 1 | - | 10 |  |  |  |  |
| 6º Passo | 2 | - | 1 | - | 10 |  |  |  |  |
| 7º Passo | 2 | - | - | - | 10 |  |  |  |  |
| 8º Passo | 2 | - | - | 2<=2 | 10 |  |  |  |  |
| 9º Passo | 2 | - | - | - | 12 |  |  |  |  |
| 10º Passo | 2 | - | - | - | 12 |  |  |  |  |
| 11º Passo | 2 | - | - | - | 12 |  |  |  |  |
| 12º Passo | 2 | - | - | 3<=2 | 12 |  |  |  |  |
| 13º Passo | 2 | - | - | - | 12 |  |  |  |  |
| 14º Passo | 2 | - | - | - | 12 |  |  |  |  |

1. Cria um programa em *javascript* que implemente o algoritmo apresentado.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **nome** | **peso** | **peso < 70** | **dif ← 70 - peso** | **Saída** |
| 1º passo |  |  |  |  |  |
| 2º passo |  |  |  |  |  |
| 3º passo |  |  |  |  |  |
| 4º passo |  |  |  |  |  |
| 5º passo |  |  |  |  |  |
| 6º passo |  |  |  |  |  |

# Exercícios de aplicação

1. Crie um algoritmo, e o respetivo programa em javascript, que permita a leitura de quatro números inteiros, determinando e apresentando para cada um deles se:

- O número é de valor par e superior a dez;

- O número é de valor ímpar e inferior a dez.

# Ciclo *While* (Enquanto)

O ciclo while é um tipo de estrutura de repetição em que o número de vezes a repetir o bloco de instruções é controlado pelo resultado da avaliação de uma condição ou expressão lógica. Enquanto o resultado da avaliação for verdadeiro (true) o bloco de instruções é executado. Uma vez que a condição ou expressão lógica é avaliada no início da estrutura, se na primeira avaliação o resultado for falso (false), o bloco de instruções não chega a ser executado uma única vez.

Este ciclo pode, também, ser implementado sabendo, inicialmente, o número de vezes que o bloco de instruções vai ser repetido. Neste caso, é utilizada uma variável contador destinada a controlar a execução do ciclo, à semelhança do funcionamento do ciclo for.

**Algoritmo conta\_nomes;**

Início

conta ← 0;

Escrever (“Digite um nome: “);

Ler (nome);

Enquanto nome <> “Fim” Fazer

conta ← conta + 1

Escrever (“Digite um nome: “);

Ler (nome);

FimEnquanto

Escrever (“Foram digitados “, conta, “ nomes.”);

Fim

1. Cria um programa em *javascript* que implemente o algoritmo apresentado.

# Ciclo *Repeat* (Enquanto)

O ciclo Repeat éum tipo de estrutura de repetição em que o número de vezes a repetir o bloco de instruções é controlado pelo resultado da avaliação de uma condição ou expressão lógica. Até que (until) o resultado da avaliação seja verdadeiro (true), o bloco de instruções continua a ser executado. Como a condição ou expressão lógica é avaliada no fim da estrutura, é sempre executado, pelo menos uma vez, o bloco de instruções.

**Algoritmo retangulo;**

Início

Repetir

Escrever (“Medidas do retângulo “);

Escrever (“Indique a largura em metros: “);

Ler (larg);

Escrever (“Indique o comprimento em metros: “);

Ler (comp);

perímetro ← 2 \* (larg + comp);

Escrever (“Perímetro = “, perímetro, “ metros.”);

Escrever (“Pretende efetuar mais cálculos (s/n)”);

Ler (resp);

Até que (resp <> “s”;

Fim

1. Cria um programa em *javascript* que implemente o algoritmo apresentado.

# Como controlar o fluxo nos ciclos de repetição?

|  |  |
| --- | --- |
| **Instrução** | **Descrição** |
| Continue | Pode ser utilizada para reiniciar uma ação nos ciclos while, do while ou for, se ocorrer uma determinada condição. |
| Break | Pode ser utilizada para interromper um while, do while, if ou switch. Termina e sai da estrutura, transferindo o controlo do programa à instrução seguinte. |