



ALGORITMOS E PROGRAMAÇÃO

Apresentação 07

Roteiro

- Apresentação dos Objetivos
- Levantamento de conhecimentos prévios
- Resolução de exercícios das aulas anteriores
- Estrutura de Repetição While (Enquanto)
- Atividade Prática

ESTRUTURA DE REPETIÇÃO: DO WHILE

Imagine que você precisa desenvolver um programa C que imprima os números pares de 0 até 4.

```
1  #include <stdio.h>
2  #include <stdlib.h>
3
4  int main(int argc, char *argv[]) {
5      int pares;
6
7      printf("Numeros Pares:\n");
8      pares = 0;
9      printf("%d\n", pares);
10     pares = 2;
11     printf("%d\n", pares);
12     pares = 4;
13     printf("%d\n", pares);
14     return 0;
15 }
```

pares
0
2
4

ESTRUTURA DE REPETIÇÃO: DO WHILE

Imagine que você precisa desenvolver um programa C que imprima os números pares de 0 até 4.

```
1 #include <stdio.h>
2 #include <stdlib.h>
3
4 int main(int argc, char *argv[]) {
5     int pares;
6
7     printf("Numeros Pares:\n");
8     pares = 0;
9     printf("%d\n", pares);
10    pares = pares + 2;
11    printf("%d\n", pares);
12    pares = pares + 2;
13    printf("%d\n", pares);
14    return 0;
15 }
```

pares
0
0 + 2 = 2
2 + 2 = 4

Contadores – Passo 2

ESTRUTURA DE REPETIÇÃO: DO WHILE

Imagine que você precisa desenvolver um programa C que imprima os números pares de 0 até 20.

```
1 #include <stdio.h>
2 #include <stdlib.h>
3
4 int main(int argc, char *argv[]) {
5     int pares;
6
7     printf("Numeros Pares:\n");
8     pares = 0;
9     printf("%d\n", pares);
10    pares = pares + 2;
11    printf("%d\n", pares);
12    pares = pares + 2;
13    printf("%d\n", pares);
14    pares = pares + 2;
15    printf("%d\n", pares);
16    pares = pares + 2;
17    printf("%d\n", pares);
18    pares = pares + 2;
19    printf("%d\n", pares);
20    pares = pares + 2;
21    printf("%d\n", pares);
22    pares = pares + 2;
23    printf("%d\n", pares);
24    pares = pares + 2;
25    printf("%d\n", pares);
26    pares = pares + 2;
27    printf("%d\n", pares);
28    pares = pares + 2;
29    printf("%d\n", pares);
30
31    return 0;
32 }
```

ESTRUTURA DE REPETIÇÃO: DO WHILE

Imagine que você precisa desenvolver um programa C que imprima os números pares de 0 até 1000.



ESTRUTURA DE REPETIÇÃO: DO WHILE

Para realizarmos uma repetição com teste no final, utilizamos a estrutura Repita, que permite que um bloco de comandos seja repetido enquanto uma condição for verdadeira.

algoritmo

C

```
repita  
  C1;  
  C2;  
  .  
  .  
  .  
  Cn;  
enquanto <condição>;
```

```
do{  
  C1;  
  C2;  
  C3;  
  ..  
  Cn;  
}while(condição);
```

ESTRUTURA DE REPETIÇÃO: DO WHILE

Imagine que você precisa desenvolver um programa C que imprima os números pares de 0 até 1000.

```
1  #include <stdio.h>
2  #include <stdlib.h>
3  #include <math.h>
4
5  int main(int argc, char *argv[]) {
6      int pares;
7      printf("Numeros Pares:\n");
8      do{
9          printf("%d \n",pares);
10         pares = pares + 2;
11     }while(pares<=1000);
12     return(0);
13 }
```

Enquanto a Condição for VERDADEIRA, a Lista de Comandos é executada.

ESTRUTURA DE REPETIÇÃO: DO WHILE

NOTA: Tradicionalmente a estrutura de repetição REPITA é descrita como REPITA – ATÉ, ou seja uma lista de comandos é repetida até que uma condição se torne verdadeira. Já na linguagem C o que existe é o REPITA- ENQUANTO, que repete uma lista de comandos enquanto uma condição é verdadeira.

ESTRUTURA DE REPETIÇÃO: DO WHILE

Crie os seguintes códigos em linguagem C usando “do while”.

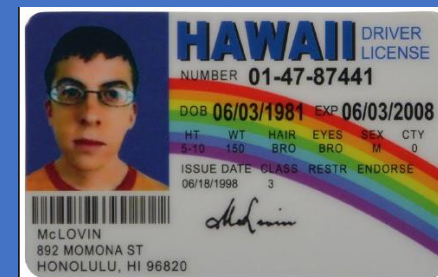
1. *Crie um programa em C que leia um número e imprima a sua tabuada.*
2. *Crie um programa em C que leia dois números pares e positivos e imprima todos os pares entre eles.*
3. *Crie um programa em C que leia dois números pares e positivos e imprima a soma de todos os pares entre eles.*
4. *Crie um programa em C que leia dois números pares e positivos e imprima a média aritmética de todos os pares entre eles.*
5. *Crie um programa em C que leia um número inteiro e imprima o fatorial desse número.
Ex.: fatorial de 5 = $5*4*3*2*1$, ou seja 120*

ESTRUTURA DE REPETIÇÃO: DO WHILE

Tanto no caso da estrutura ENQUANTO, quanto na estrutura REPITA- ENQUANTO, o número de repetições pode não ser conhecido. Ex.: Repetir a digitação até que seja informado um CPF válido ou idade positiva.

Crie um programa em C que leia a idade de uma pessoa.

- Valide a idade: Repetir a leitura da idade enquanto for digitada idade menor que zero.
- Se a idade for maior ou igual a 18 mostre a mensagem: "Você pode votar e dirigir"
- Se a idade for maior ou igual a 16 mostre a mensagem: "Você pode dirigir"
- Se a idade for menor que 16 mostre a mensagem: "Você é menor"



Teste de Mesa

[illegible]

ESTRUTURA DE REPETIÇÃO: DO WHILE

```
1 #include <stdio.h>
2 #include <stdlib.h>
3
4 int main(int argc, char *argv[]) {
5     int idade;
6
7     do{
8         printf("\nInforme a idade: ");
9         scanf("%d",&idade);
10    }while(idade<0);
11
12    if(idade>=18){//opcional
13        printf("\nVoce pode votar e dirigir");
14    }//opcional
15    else
16    {
17        if(idade>=16)
18        { //opcional
19            printf("\nVoce pode votar");
20        }//opcional
21        else
22        { //opcional
23            printf("\nMenor de idade");
24        }//opcional
25    }
26    return 0;
27 }
```

Teste de Mesa

idade	Tela
-5	Informe a idade:
-3	Informe a idade:
-10	Informe a idade:
-7	Informe a idade:
	Informe a Idade
8	Menor de idade
	Informe a Idade
15	Menor de idade
	Informe a idade
16	Você pode votar
	Informe a idade
21	Você pode votar e dirigir

ESTRUTURA DE REPETIÇÃO: FOR

Utilizamos a estrutura de repetição FOR (PARA) quando sabemos exatamente o número de repetições. A estrutura FOR é dita “estrutura de controle de repetição com variável de controle” pois através dessa variável possui limites fixos para as repetições.

ESTRUTURA DE REPETIÇÃO: FOR

Para realizarmos uma repetição com variável de controle, utilizamos a estrutura PARA, que permite que um bloco de comandos seja repetido de um valor inicial(vi) até um valor final(vf) para a variável de controle.

algoritmo

```
para V de vi até vf passo p faça  
    C1;  
    C2;  
    .  
    .  
    .  
    Cn;  
fimpara;
```

C

```
for(ValorInicial;condição;incremento)  
{  
    C1;  
    C2;  
    C3;  
    ...  
    Cn;  
}
```

ESTRUTURA DE REPETIÇÃO: FOR

Imagine que você precisa desenvolver um programa C que imprima os números pares de 0 até 1000.

```
1 #include <stdio.h>
2 #include <stdlib.h>
3 #include <math.h>
4
5 int main(int argc, char *argv[]) {
6     int pares;
7     printf("Numeros Pares:\n");
8     for(pares=0;pares<=1000;pares+=2)
9     {
10         printf("%d \n",pares);
11     }
12     return(0);
13 }
```

A Lista de Comandos é executada 501 vezes. (começa no zero)

ESTRUTURA DE REPETIÇÃO: FOR

Crie os seguintes códigos em linguagem C usando “for”.

1. *Crie um programa em C que leia a altura e o sexo de 20 alunos e informe:*
 - a) *Número de alunos*
 - b) *Número de alunas*
 - c) *Média de altura dos alunos*
 - d) *Média de altura das alunas*

2. *O Campeonato Brasileiro conta com 20 times na série A no ano de 2018. Sabendo que cada time disputa 38 rodadas e que vitórias contabilizam 3 pontos, empates 1 ponto e derrotas 0 pontos, crie um programa na linguagem C que:*
 - a) *Leia o nome do time e números de vitórias, empates e derrotas dos 20 times e calcule e imprima sua pontuação e aproveitamento*
 - b) *Mostre o nome e pontuação do Campeão e do Vice-Campeão*
 - c) *Mostre a diferença de pontos entre Campeão e Vice-Campeão*
 - d) *Mostre o nome e aproveitamento do penúltimo e último colocados*
 - e) *Mostre a diferença de pontos entre penúltimo e último colocados*



ecosistema
ănimă