

Programação de Computadores IV

Estruturas de repetição

Slides cedidos por Bruno Augusto Dorta Marques

Estruturas de repetição

- Uma ação que se repete várias vezes pode ser tratada através:
 - Repetição de código, escrever várias vezes o mesmo código
 - Utilizando uma estrutura de repetição
- Exemplo:
 - Faça um programa que escreve na tela “Olá mundo” 20 vezes.

Exemplo

- Faça um programa que escreve na tela “Olá mundo” 20 vezes.
- Repetição de código não é aconselhável.
- Como devemos escrever se o problema pedir para escrever 1 milhão de vezes ?

```
int main(void){  
  
    printf("Ola Mundo\n");  
    printf("Ola Mundo\n");  
    printf("Ola Mundo\n");  
    printf("Ola Mundo\n");  
    printf("Ola Mundo\n");  
    printf("Ola Mundo\n");  
    printf("Ola Mundo\n");  
    printf("Ola Mundo\n");  
    printf("Ola Mundo\n");  
    printf("Ola Mundo\n");  
    printf("Ola Mundo\n");  
    printf("Ola Mundo\n");  
    printf("Ola Mundo\n");  
    printf("Ola Mundo\n");  
    printf("Ola Mundo\n");  
    printf("Ola Mundo\n");  
    printf("Ola Mundo\n");  
    printf("Ola Mundo\n");  
    printf("Ola Mundo\n");  
  
    return 0; }
```

Estrutura Enquanto

Enquanto expressão lógica verdadeira faça

<bloco de comando>

Fim enquanto

SINTAXE:

```
while( expressão-lógica == true){  
    bloco de comando  
}
```

Exemplo

- Faça um programa que escreve na tela “Olá mundo” 20 vezes.

```
Int main(void){  
  
    int contador = 0;  
    while( contador < 20){  
        printf("Ola mundo\n");  
        contador = contador + 1;  
    }  
    return 0;  
}
```

Exercícios

- Faça um programa que:
 - leia N números inteiros do teclado.
 - Ao final mostre qual é o maior número inserido

Estrutura Enquanto

- É comum utilizarmos uma variável como contador de iterações.
- O contador é um número inteiro. Inicializado com 0.
- A cada iteração, o contador é incrementado em 1
- Podemos utilizar o contador em conjunto com o enquanto para repetir uma instrução n vezes.

SINTAXE:

```
int contador = 0;
while( contador < 10){
    printf("essa é uma iteração");
    contador = contador + 1;
}
```

Estrutura Para

Para variável = inicio, até fim, com incremento repetir

<bloco de comando>

Fim para

SINTAXE:

```
for( var = inicio; expressão lógica contendo var; var = var + 1){  
    bloco de comando  
}
```

Exemplo

- Faça um programa que escreve na tela “Olá mundo” 20 vezes.

```
Int main(void){  
  
    int contador = 0;  
    for( contador = 0; contador < 20; contador = contador + 1){  
        printf("Olá Mundo\n");  
    }  
    return 0;  
}
```

Estrutura para

- A variável utilizada como contador deve ser declarada no início do programa
- É obrigatório informar o valor inicial da variável na estrutura para
- A condição de repetição normalmente envolve o contador
- O incremento pode ser alterado

```
Int main(void){  
  
    int contador = 0;  
    for( contador = 0; contador < 20; contador = contador + 1){  
        printf("Olá Mundo\n");  
    }  
    return 0;  
}
```

Equivalência Enquanto – Para

- As estruturas para e enquanto são equivalentes:

```
Int main(void){  
  
    int contador = 0;  
    while( contador < 20){  
        printf("Olá Mundo\n");  
        contador = contador + 1;  
    }  
    return 0;  
}
```

```
Int main(void){  
  
    int contador = 0;  
    for( contador = 0; contador < 20; contador = contador + 1){  
        printf("Olá Mundo\n");  
    }  
    return 0;  
}
```

Incremento

- O Incremento pode ser alterado.
- É possível escrever o incremento de uma forma compacta:

é o mesmo que

```
contador = contador + 1;  
contador++;
```

```
contador = contador + 5;  
contador += 5;
```

```
Int main(void){  
  
    int contador = 0;  
    for( contador = 0; contador < 20; contador++){  
        printf("%d\n", contador);  
    }  
  
    return 0;  
}
```

Exercícios

- Escreva o código abaixo, substituindo a estrutura **enquanto** pela estrutura **para**:

```
Int main(void){  
  
    int contador = 0, x, y;  
    x = 1;  
    y = 0  
    while( contador < 20){  
        x = 2 * contador;  
        y = y + x - 1;  
        contador = contador + 1;  
        y = - y;  
    }  
    return 0;  
}
```

Exercícios

- Escreva o código abaixo, substituindo a estrutura **enquanto** pela estrutura **para**:

```
Int main(void){  
  
    int contador = 0, x, y;  
    x = 1;  
    y = 0  
    while( contador < 20){  
        x = 2 * contador;  
        y = y + x - 1;  
        contador = contador + 1;  
        y = -y;  
    }  
    return 0;  
}
```

```
Int main(void){  
  
    int contador = 0, x, y;  
    x = 1;  
    y = 0  
    for( contador = 0; contador < 20; contador++){  
        x = 2 * contador;  
        y = y + x - 1;  
        y = -y;  
    }  
    return 0;  
}
```

Exercícios

- Faça um programa que imprima todos os números pares de Zero a 100:

Exercícios

- Faça um programa que imprima todos os números pares de Zero a 100:

```
Int main(void){  
  
    int contador= 0;  
    for( contador= 2; contador<= 100 ; contador+=2){  
        printf("%d\n", contador);  
    }  
  
    return 0;  
}
```

Exercícios

- Faça um programa que imprima todos os números ímpares de 1 a 100 em ordem decrescente:

Exercícios

- Faça um programa que imprima todos os números ímpares de 1 a 100 em ordem decrescente:

```
Int main(void){  
  
    int contador = 1;  
    for( contador = 99; contador > 0 ; contador -= 2){  
        printf("%d\n", contador);  
    }  
  
    return 0;  
}
```

Exercícios

- Faça um programa que imprima a tabuada do 5.

Exercícios

- Faça um programa que imprima a tabuada do 5.

```
Int main(void){  
  
    int contador, x;  
    for( contador = 0; contador <= 10 ; contador++ ){  
        x = 5* contador;  
        printf("5* %d = %d\n", contador, x);  
    }  
  
    return 0;  
}
```

- Faça um programa que:
 - leia N números inteiros do teclado.
 - Ao final mostre qual é o maior número inserido

Exercícios

```
Int main(void){  
  
    int contador = 0, N, maior = -99999999, temp;  
    printf("Insira a quantidade de números a serem lidos:\n");  
    scanf("%d",&N);  
    while( contador < 20){  
        printf("\nInsira um número inteiro:\n");  
        scanf("%d",&temp);  
        if(temp > maior){  
            maior = temp;  
        }  
        contador = contador + 1;  
    }  
    printf("O maior número foi : %d \n", maior);  
    return 0;  
}
```