

Algoritmos e Estrutura de Dados I

Comandos de Repetição

prof. Frederico Santos de Oliveira

Universidade Federal de Mato Grosso
Instituto de Engenharia

Agenda

- 1 Objetivos
- 2 Comando For
- 3 Exemplos
- 4 Comando Break
- 5 Comando Continue

Objetivos

- Aprender a utilizar o comando de repetição **for**.
- Aprender a utilizar os comandos de interrupção dos laços de repetição (**break** e **continue**).
- Fazer um programa utilizando comando de repetição e interrupção.

Comando For

- O comando **for** é útil quando se conhece previamente o número exato de vezes que se deseja executar um determinado conjunto de comandos.
- Sua estrutura possui mecanismos para contar o número de vezes que o corpo do laço é executado.

Comando While

A forma do comando **while** na linguagem C é a seguinte:

```
inicializacao;  
  
while ( condicao ) {  
    conjunto de instrucoes; // Comandos a serem repetidos  
  
    incremento;  
}
```

Comando For

A forma do comando **for** na linguagem C é a seguinte:

```
for (inicializacao; condicao; incremento) {  
    conjunto de instrucoes; // Comandos a serem repetidos  
}
```

Comando For

Exemplo: mostrar os n primeiros números em ordem crescente.

Comando While

Solução:

```
#include<stdio.h>
int main() {
    int i, n;
    printf("Informe o valor de n:");
    scanf("%d", &n);
    for (i = 0; i < n; i++) {
        printf("%d\n",i);
    }
    return 0;
}
```

Comando For

Exemplo: mostrar os n primeiros números em ordem decrescente.

Comando For

Solução:

```
#include<stdio.h>
int main() {
    int i, n;
    printf("Informe o valor de n:");
    scanf("%d", &n);
    for (i = n; i > 0; i--) {
        printf("%d\n",i);
    }
    return 0;
}
```

Comando For

Exemplo: mostrar os n primeiros números pares.

Comando For

Solução 1:

```
#include<stdio.h>
int main() {
    int i, n;
    printf("Informe o valor de n:");
    scanf("%d", &n);
    for (i = 0; i < n; i++) {
        printf("%d\n",2*i);
    }
    return 0;
}
```

Comando For

Exemplo: programa que calcula a soma dos valores digitados pelo usuário. A finalização da entrada de dados é encerrada quando o usuário digitar -1.

Comando For

Solução:

```
#include<stdio.h>
int main() {
    int i, n, soma = 0, valor = 0;
    for (i = 0; valor!=-1; i++) {
        printf("Digite um valor ou -1:");
        scanf("%d",&valor);
        if (valor != -1) {
            soma+=valor;
        }
    }
    printf("Soma = %d\n",soma);
    return 0;
}
```

Comando For

Exemplo: programa que calcula a média dos valores digitados pelo usuário. A finalização da entrada de dados é encerrada quando o usuário digitar -1.

Comando For

Solução:

```
\begin{lstlisting}
#include<stdio.h>
int main() {
    int i, n, soma = 0, valor = 0;
    for (i = 0; valor!=-1; i++) {
        printf("Digite um valor ou -1:");
        scanf("%d",&valor);
        if (valor != -1) {
            soma+=valor;
        }
    }
    // Cuidado ao calcular a media
    printf("Média = %d\n",soma/(i-1)); // Necessario subtrair, pois i contou o
        valor -1
    return 0;
}
```


Comando For

Exemplo: programa que dado um valor n fornecido pelo usuário, informa $n!$.

Comando For

Solução:

```
#include<stdio.h>
int main() {
    int i, n, fat = 1;
    printf("Informe o valor de n:");
    scanf("%d",&n);
    for (i = 1; i <= n; i++) {
        fat*=i; // Equivalente a fat=fat*i
    }
    printf("n!=%d",fat);
    return 0;
}
```

Comando For

Exemplo: programa que, dado o valor de n fornecido pelo usuário, calcula o valor da série $1 + \frac{1}{2} + \frac{1}{3} + \dots + \frac{1}{n}$

Comando For

Solução:

```
#include<stdio.h>
int main() {
    int i, n;
    float soma = 0;
    printf("Informe o valor de n:");
    scanf("%d",&n);
    for (i = 1; i <= n; i++) {
        soma+=1.0/i;
    }
    printf("soma=%f",soma);
    return 0;
}
```

Comando For

Exemplo: programa que, dado o valor de n fornecido pelo usuário, calcula o valor da série

$$1 - \frac{1}{4} + \frac{1}{9} - \frac{1}{16} \dots + \frac{1}{n^2}$$

Comando For

Solução 1:

```
#include<stdio.h>
int main() {
    int i, n;
    float soma = 0;
    printf("Informe o valor de n:");
    scanf("%d",&n);
    for (i = 1; i <= n; i++) {
        if (i%2==0) {
            soma-=1.0/(i*i);
        }
        else {
            soma+=1.0/(i*i);
        }
    }
    printf("soma=%f",soma);
    return 0;
}
```

Comando For

Solução 2:

```
#include<stdio.h>
int main() {
    int i, n;
    float soma = 0, sinal = 1;
    printf("Informe o valor de n:");
    scanf("%d",&n);
    for (i = 1; i <= n; i++) {
        soma+=sinal*1.0/(i*i);
        sinal*=-1;
    }
    printf("soma=%f",soma);
    return 0;
}
```

Comando Break

- Usando o comando **break**, podemos interromper um *loop* mesmo que a condição ainda seja verdadeira.
- Pode ser usado para terminar um *loop* infinito.
- Por exemplo, podemos interromper uma contagem antes de seu final.

Comando Break

Exemplo:

```
#include<stdio.h>
int main() {
    int i, n;
    for (i = 1; i <= 10; i++) {
        printf("%d ",i);
        if (i==5){
            printf("Contagem cancelada!");
            break;
        }
    }
    return 0;
}
```

Saída: 1 2 3 4 5 Contagem cancelada!

Comando Break

Exemplo: programa que, dado o valor de n fornecido pelo usuário, verifica se n é primo.

Comando Break

Solução:

```
#include<stdio.h>
int main() {
    int i, n;
    printf("Informe o valor de n:");
    scanf("%d",&n);
    for (i = 2; i < n; i++) {
        if (n%i==0) {
            printf("Nao eh primo");
            break;
        }
    }
    if (i==n) {
        printf("Eh primo");
    }
    return 0;
}
```

Comando Continue

- O comando **continue** faz com que uma iteração seja saltada.
- É como se o fim do bloco tivesse sido alcançado
- O programa salta para a próxima iteração.
- Por exemplo, saltaremos o número 5 na contagem regressiva.

Comando Continue

Exemplo:

```
#include<stdio.h>
int main() {
    int i, n;
    for (i = 1; i <= 10; i++)
        if (i==5){
            continue;
        }
        printf("%d ",i);
    }
    return 0;
}
```

Saída: 1 2 3 4 6 7 8 9 10

Comando Continue

Exemplo: programa que, dado o valor de n fornecido pelo usuário, informe todos os divisores de n .

Comando Continue

Solução:

```
#include<stdio.h>
int main() {
    int n, div;
    printf("Informe o valor de n:");
    scanf("%d",&n);
    printf("Os divisores de %d sao: ", n);
    for (div = 1; div < n; div++) {
        if (n%div != 0) { /* Testa se nao eh divisor */
            continue;
        }
        printf("%d ",div); // Imprime o divisor
    }
    printf("%d\n.", n); /* n eh o ultimo divisor */
    return 0;
}
```

Fim

Fim