

Funções e Procedimentos

Prof. Tiago Gonçalves Botelho

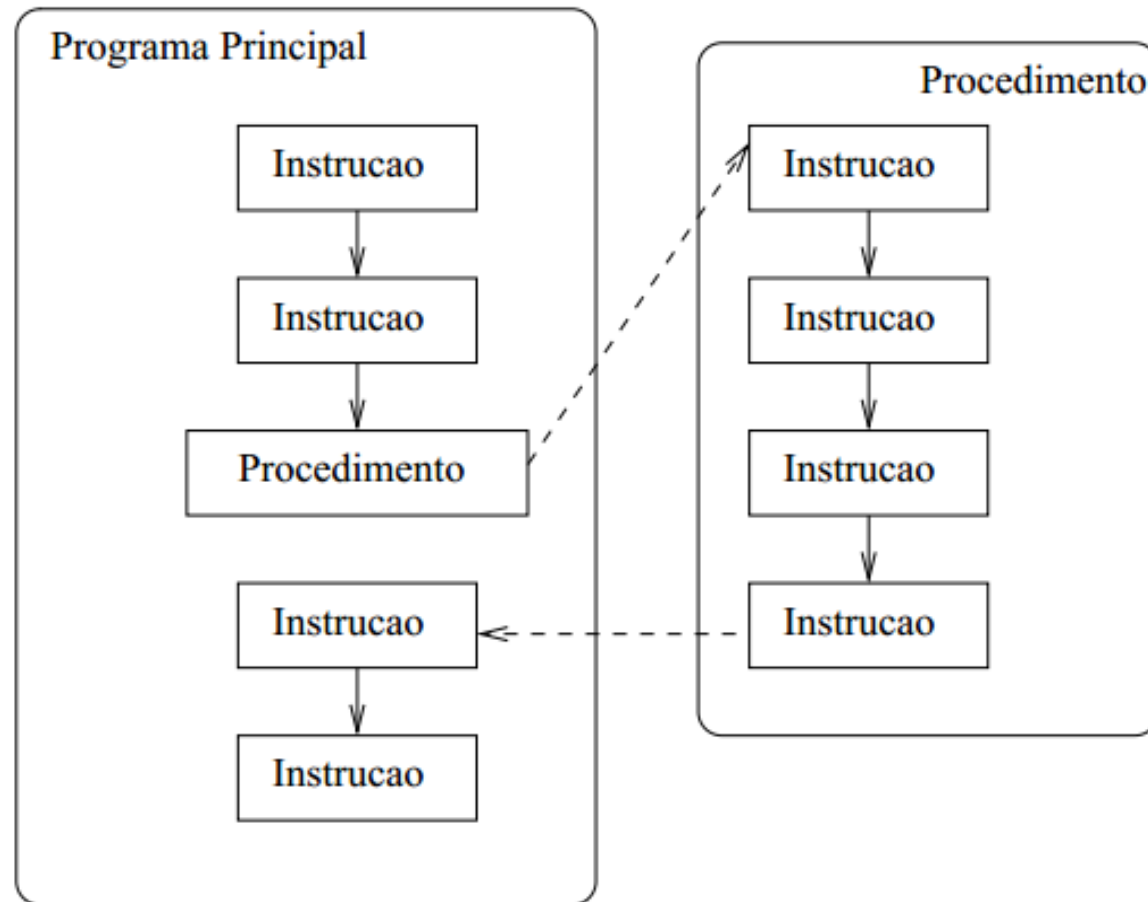
Funções e Procedimentos (Modularização)

- A modularização de um algoritmo/programa é uma forma de dividir as tarefas em subalgoritmos/subprogramas, e cada um desses módulos cuida de uma parte separada do problema.

Motivação para o uso de Funções e Procedimentos

- Permitir o reaproveitamento de código já construído (por você ou por outros programadores);
- Evitar que um trecho de código que seja repetido várias vezes dentro de um mesmo programa;
- Permitir a alteração de um trecho de código de uma forma mais rápida. Com o uso de uma função é preciso alterar apenas dentro da função que se deseja;
- Evitar que os blocos do programa fiquem grandes demais e, por consequência, mais difíceis de entender;

Fluxo de execução de um programa usando Procedimento



Definição de procedimentos e funções

- **Procedimentos** são estruturas que agrupam um conjunto de comandos, que são executados quando o procedimento é chamado.
- **Funções** são semelhantes aos procedimentos, exceto que uma função sempre retorna um valor.

Funções e Procedimentos

- Funções que não retornam valor (**Procedimentos**)
 - Sem parâmetro
 - Com parâmetro
- Funções que retornam valor (**Funções**)
 - Sem parâmetro
 - Com parâmetro
- Variável global (pode ser utilizada em todo o programa) e Variável local (só é válida dentro do procedimento ou função)

Funções que não retornam valor e sem parâmetro

- Exemplo 01: Faça um programa que receba dois valores, some os dois números em uma função e exiba o resultado na função principal.

Funções que não retornam valor e sem parâmetro

```
1  #include<stdio.h>
2
3  int a,b,s;
4  void soma(void);
5
6  main() {
7      printf("Entre com um valor: ");
8      scanf("%i",&a);
9      printf("Entre com outro valor: ");
10     scanf("%i",&b);
11     soma();
12     printf("A soma e: %i",s);
13 }
14
15 void soma(void) {
16     s=a+b;
17 }
```

→ Não retorna valor

→ Sem parâmetro

Funções que não retornam valor e com parâmetro

- Exemplo 02: Faça um programa usando procedimento que receba um valor representando os segundos. Este procedimento deverá convertê-lo para horas, minutos e segundos.

Funções que não retornam valor com parâmetro

```
1  #include<stdio.h>
2
3  void transformacao(int);
4
5  main(){
6      int seg;
7      printf("Entre com os segundos: ");
8      scanf("%i",&seg);
9      transformacao(seg);
10 }
11
12 void transformacao(int segundos){
13     int h,m,s,r;
14     h=segundos/60;
15     r=segundos%3600;
16     m=r/60;
17     s=r%60;
18     printf("\n Hora=%i \n Minutos=%i \n Segundos=%i",h,m,s);
19 }
```

Não retorna valor

Com parâmetro

Funções que retornam valor e sem parâmetro

- Exemplo 03: Faça um algoritmo que utilize uma função para calcular a multiplicação de dois números, ela não receberá nenhum valor como parâmetro. Mostre o resultado no programa principal.

Funções que retornam valor e sem parâmetro

```
1  #include<stdio.h>
2
3  int multiplica(void);
4  int x,y;
5
6  main(){
7      int res;
8      printf("Entre com um valor: ");
9      scanf("%i",&x);
10     printf("Entre com outro valor: ");
11     scanf("%i",&y);
12     res=multiplica();
13     printf("\n O resultado e: %i",res);
14 }
15
16 int multiplica(void){
17     int r;
18     r=x*y;
19     return r;
20 }
```

Retorna valor

Sem parâmetro

Funções que retornam valor e com parâmetro

- Exemplo 04: Alterar o exercício anterior para passar como parâmetro os dois valores a serem multiplicados e retornar valor.

Funções que retornam valor e com parâmetro

```
1  #include<stdio.h>
2
3  int multiplica(int,int);
4
5  main() {
6      int x,y,res;
7      printf("Entre com um valor: ");
8      scanf("%i",&x);
9      printf("Entre com outro valor: ");
10     scanf("%i",&y);
11     res=multiplica(x,y);
12     printf("\n O resultado e: %i",res);
13 }
14
15 int multiplica(int a, int b){
16     int r;
17     r=a*b;
18     return r;
19 }
```

Retorna valor

Com parâmetros

Referências Bibliográficas

- ASCENCIO, A. F. G.; CAMPOS, E. A. V. Fundamentos da programação de computadores. Editora Prentice-Hall, 2008.
- MEDINA, M.; FERTIG, C. Algoritmos e Programação: teoria e prática. 2ª ed. São Paulo: Editora Novatec, 2006.