



Linguagens de Programação
Prof. Andrei Rimsa Álvares

Lista de Exercícios VI

Subprogramas

Exercício 01) Considere o trecho de código a seguir com sintaxe semelhante a C:

```
01: int x = 1, y = 2;
02:
03: void func(int a, int b) {
04:     a = 1;
05:     b = a + b;
06:     printf("x = %d e y = %d", x, y);
07:     x = (x > a) ? 6 : b;
08:     printf("x = %d e y = %d", x, y);
09: }
10: void main() {
11:     int x = 10, y = 20;
12:     printf("x = %d e y = %d", x, y);
13:     func(y, x + y);
14:     func(x, y);
15: }
```

Considerando **escopo estático**, execute o código acima considerando passagem de parâmetros por:

- Valor
- Nome (em todas as variáveis possíveis)
- Valor-resultado (em todas as variáveis possíveis)

Agora considere **escopo dinâmico** e passagem de parâmetro por:

- Valor
- Nome (em todas as variáveis possíveis)
- Valor-resultado (em todas as variáveis possíveis)

Exercício 02) Considere código seguinte com sintaxe semelhante a C++:

```
01: int x = 3, y = 4;
02: void f(int y) {
03:     y = y + x;
04:     cout << x << ' ' << y;
05: }
06: void g() {
07:     int x = 5;
08:     cout << x << ' ' << y;
09:     f(x);
10:     cout << x << ' ' << y;
11: }
12: }
13: int main() {
14:     int x = 1;
15:     int y = 2;
16:     cout << x << ' ' << y;
17:     f(x);
18:     g();
19:     cout << x << ' ' << y;
20:     return 0;
21: }
```

Execute esse código supondo:

- Escopo estático** e passagem de parâmetro por **valor-resultado**
- Escopo estático** e passagem de parâmetro **por nome**
- Escopo dinâmico** e passagem de parâmetro **por valor**
- Escopo dinâmico** e passagem de parâmetro **por referência**