

Lista de Exercícios No. 3

Para os próximos exercícios, considere uma memória de 2GB, inteiros ocupando 2 bytes e apontadores ocupando 4 bytes, registros de ativação em pilha contendo parâmetros, elo de controle e dados locais.

1. Dado o programa abaixo, diga o que será escrito se a passagem de parâmetros for feita (a) por valor e (b) por referência. Mostre a construção do mapa de memória.

```
Prog principal;  
|  Var a,b,c:inteiro;  
|  
|  Proc inic(b,c,d);  
|  Inicio  
|  |  d:=3; c:=b; b:=6; a:=a+d;  
|  Fim;  
|  
|  Proc P(b,c,d);  
|  Inicio  
|  |  a:=a+d; c:=d*b; d:=c+b-1;  
|  Fim;  
|  
Inicio  
|  a:=2; b:=1; c:=0;  
|  inic(a,b,c);  
|  P(c,b,a);  
|  Escreva(a,b,c);  
Fim.
```

2. Dado o programa abaixo, diga o que será escrito considerando:

- (a) passagem por valor
- (b) passagem por referência

```
Prog Principal;
|   Var x,y,z:byte;
|       p,q:*byte;
|
|   Proc A(y:byte);
|   Inicio
|   |   x:=y+z; y:=4;
|   Fim;
|
|   Proc B(x:byte; z:*byte);
|   Inicio
|   |   *z:=5;
|   |   New(p);
|   |   *p:=x;
|   Fim;
|
|   Proc C(q:*byte);
|   |   Var r:*byte;
|   |       z:byte;
|   Inicio
|   |   z:=3;
|   |   A(x);
|   |   New(r);
|   |   *r:=x;
|   |   Free(q); /* Mostre aqui o mapa de memoria */
|   |   p:=r;
|   Fim;
|
|   Inicio
|   |   x:=0; y:=1; z:=2;
|   |   p:=&x; q:=&y;
|   |   B(y,q);
|   |   C(p);
|
|   Escreva(x,y,z,p,q,*p,*q);
Fim.
```

3. Dado os códigos intermediários abaixo, proceda a alocação de registradores para os temporários. Após a alocação, qual seria o código ASSEMBLY 8086 gerado? Considere registradores de 8 bits.

a)

```
1:      t6 = 0;
2:      t1 = t7;
3:      t2 = t6 * 4;
4:      t3 = t1 + t2;
5: L1:  if (t6 > 100) goto L2
6:      t3 = t1 + 1;
7:      t6 = t6 + 1;
8:      t2 = t6 * 4;
9:      t5 = t7;
10:     if (t3 <= t7) goto L3
11:     t3 = t5 - t2;
12:     t1 = t5;
13:     goto L2
14: L3:  t4 = t7;
15:     t3 = t4 + t2;
16:     t1 = t4;
17: L2:
```

b) Para este exercício, i e x são variáveis do tipo byte e *t é o conteúdo apontado por t:

```
01:     t1 = 8
02:     t2 = i + 10
03:     t3 = t1 + t2
04:     t4 = *t3
05:     t5 = t4 * x
06:     t6 = i + 20
07:     t7 = t6 * 8
08:     t8 = *t7
09:     t9 = t5 + t8
10:     *t7 = t9
```