

- 1) Converta as linhas de instruções escritas em linguagem C para instruções em Assembly do MIPS. Considere que:

– As variáveis indicadas $A = 10$, $B = 15$, $C = 20$, $D = 25$, $E = 30$ e $F = 35$ foram todas armazenadas previamente na memória de dados. Use os registradores de $\$s0$ até $\$s5$ para manipular os dados.

– Os vetores G e H possuem quatro posições cada e inicializam com zeros em suas posições (words). Use os registradores $\$s6$ e $\$s7$ para manipular estas posições.

– Implemente as linhas de instruções abaixo como sendo linhas de um mesmo programa.

a) $G[0] = (A - (B + C) + F);$

b) $G[1] = E - (A - B) * (B - C);$

c) $G[2] = G[1] - C;$

d) $G[3] = G[2] + G[0];$

e) $H[0] = B - C;$

f) $H[1] = A + C;$

g) $H[2] = B - C + G[3];$

h) $H[3] = B - G[0] + D;$

- 2) Faça um programa que calcule: $\sum_{i=1}^5 i$

- 3) Faça um programa que carregue o vetor $v=[1,3,2,1,4,5]$ para a memória. Ou seja, que contenha a sequência de instruções MIPS correspondentes às instruções C seguintes:

$v[0] = 1;$

$v[1] = 3;$

$v[2] = 2;$

$v[3] = 1;$

$v[4] = 4;$

$v[5] = 5;$

Assuma que o registrador $\$s0$ contém o endereço de início do vetor como sendo $0x10010020$.

- 4) Modifique o programa anterior fazendo o acesso à memória de forma indexada. Isto é, assumindo que o registrador $\$s0$ contém o valor $0x10010020$, faça um programa que contenha a sequência de instruções MIPS correspondentes às instruções C seguintes:

```

i = 0;
v[i] = 1;
i = i + 1;
v[i] = 3;
i = i + 1;
v[i] = 2;
i = i + 1;
v[i] = 1;
i = i + 1;
v[i] = 4;
i = i + 1;
v[i] = 5;

```

5) Reproduza cada uma das expressões escritas em linguagem de alto nível a seguir para Assembly do MIPS. Considerações:

– As variáveis **a** e **b** são posições de memória (segmento de dados). Os valores (inteiros positivos) dessas variáveis devem ser informados pelo usuário via teclado.

– Implemente pequenos programas, um para cada expressão abaixo.

a) if (a > b)

```

    a = a + 1;

```

b) if (a ≥ b)

```

    b = b + 1;

```

c) if (a ≤ b)

```

    a = a + 1;

```

d) if (a == b)

```

    b = a;

```

e) if (a < b)

```

    a = a + 1;

```

```

    else

```

```

        b = b + 1;

```

f) a = 0;

```

    b = 0;

```

```

    c = 5;

```

```

    while (a < c){

```

```

        a = a + 1;

```

```

        b = b + 2;}

```

```
g) a = 1;
    b = 2;
    for (i = 0; i < 5; i ++){
        a = b + 1;
        b = b + 3;}
```

```
h) switch(a){
    case 1:
        b = c + 1;
        break;
    case 2:
        b = c + 2;
        break;
    default:
        b = c;
        break;}
```