

Arquitetura e Sistemas de Computadores I

Assembly

1. Usando as instruções `add`, `addi`, `sll` e `sub`, escreva programas em assembly para as seguintes expressões:

- (a) $t0 = 3 + 4$. (use instruções para colocar valores nos registros)
(b) Supondo ordenação de bytes little endian, indique (desenhando o mapa de memória) qual o conteúdo de cada endereço de memória:

Address	Memory
\vdots	\vdots
0x00400002	??
0x00400001	??
0x00400000	??

- (c) $t0 = 3 + 4 + 5$. (use instruções para colocar valores nos registros)
(d) Supondo ordenação de bytes big endian, indique (desenhando o mapa de memória) qual o conteúdo de cada endereço de memória. Qual o byte no endereço 0x00400011? e no endereço 0x0040000b?
(e) $t1 = 2 \times t0 + 1$.
(f) $t1 = 9 \times t0 + 5$.
(g) $t1 = t0 - 1$. (haverá necessidade de uma instrução `subi`?)
(h) $t2 = 2 \times t0 + 3 \times (t0 - t1)$.
2. Escreva um troço de código (em papel) para calcular o simétrico de um número guardado no registro `$t0`. O resultado deve ficar em `$t1`.
3. Sem usar o simulador, analise o seguinte troço de código e indique o valor final dos registros `$t0`, `$t1` e `$t2`.

```
lui $t0, 0x1234
ori $t0, $t0, 0x5678
ori $t1, $zero, 0x1234
sll $t1, $t1, 16
ori $t1, $t1, 0x5677
sub $t2, $t1, $t0
```

4. Sem usar o simulador, analise o seguinte troço de código e indique os valores finais de `$t0` e `$t1`.

```
lui $t0, 0x0001
ori $t0, $t0, 0xffff
lui $t1, 0x0002
ori $t1, $t1, 0x0ff7

or $t2, $zero, $t0
or $t0, $t1, $zero
or $t1, $t2, $t2
```

5. Supondo que $\$t0=0x00107fff$ e $\$t1=0x80000000$, determine o resultado das seguintes operações:
- (a) `sra $t2, $t0, 4`
 - (b) `srl $t2, $t0, 2`
 - (c) `sra $t2, $t1, 4`
 - (d) `srl $t2, $t1, 2`
6. Suponha que o programa da alínea 3 está carregado em memória no endereço $0x00400000$ e que os registos do processador têm os valores $\$t0=0xffffffff$, $\$t1=0x80000000$ e $PC = 0x00400010$. Qual o conteúdo dos registos todos ($\$t0$, $\$t1$, $\$t2$ e PC) após a execução de duas instruções.