

- 1) Escreva um pequeno programa que armazene as seguintes *half-words*: 1, 2, -2, -3. Verifique na memória de dados como estes números foram armazenados.
- 2) Implemente o código abaixo e verifique o resultado final nos registradores \$t0 e \$t1:

```
1  .data
2      ENTRADA: .byte 1 2 -2 -3 -4
3  .text
4  main:
5      la      $s2, ENTRADA
6      lb      $t0, 2($s2)
7      lbu     $t1, 3($s2)
8  loop:
9      j       loop
```

- 3) Escreva um programa em Assembly do MIPS que receba uma *string* do teclado com até 7 dígitos e armazene esta *string* recebida na memória de dados.

Sugestão: use o comando **8** na chamada de sistema **syscall**.

- 4) Adapte o programa anterior para contar quantas letras **a** (minúsculas) existem na palavra armazenada.

Sugestão: teste o programa digitando a palavra **abacate** e veja se o seu programa contará certo a quantidade de letras **a**.

- 5) Adapte o programa do item 4) para utilizar uma estrutura de iteração (laço **for**) para testar as letras da *string* armazenada.
- 6) Adapte o programa do item 4) para que utilize uma estrutura de iteração tipo **while**, com critério de parada ao encontrar o final da *string* armazenada usando o caractere **\0 (NULL)**.
- 7) Considere que o processador MIPS está sendo utilizado para controlar um conjunto de 32 dispositivos externos. Cada dispositivo recebe um bit indicando se ele deve ser ligado (=1) ou desligado (=0). Estes bits estão armazenados no endereço de dados **0x10010020**. Implemente um programa que faça as seguintes operações:

- O usuário deve informar pelo teclado se quer ligar ou desligar um bit. **Exemplo:** se digitar **1**, é para ligar algum bit. Se digitar **0**, é para desligar algum bit;
- O usuário deve informar, na sequência, qual bit deve ser ligado/desligado. **Por exemplo**, ao digitar **2**, então o bit 2 do endereço 0x10010020 deve ser ligado ou desligado (conforme instrução anterior).
- Os estados anteriores dos outros bits devem ser preservados obrigatoriamente.