

**Parte Teórica**

- 1** Considere uma máquina com uma cache com mapeamento direto, blocos de 1 Byte e com 16 bits de *tag*. Esta máquina tem uma memória RAM com 1GB de capacidade.
  - a) Qual a função da *tag*? Justifique. **(0,5 Val.)**
  - b) Qual a função dos *valid bits*? Justifique. **(0,5 Val.)**
  - c) Qual a capacidade total desta cache, contando com os bits da *tag* mais os *valid bits*? **(2 Val.)**
  
- 2** O tipo mais comum de discos rígidos é o disco *Winchester* composto por uma unidade selada com um conjunto de elementos dentro de uma caixa de metal. Considere um disco com capacidade de 32 GB com pratos de dupla face, oito sectores por pista, onde cada sector armazena 512 bytes em cada pista. Este disco tem a capacidade de armazenar 32KB por cilindro. Calcule:
  - a) O número de cilindros deste disco. **(1 Val.)**
  - b) A capacidade de cada pista. **(1 Val.)**
  
- 3** Qual a principal diferença, em termos estruturais, das memórias estáticas (SRAM) relativamente às memórias dinâmicas (DRAM)? Apresente as vantagens e desvantagens de umas relativamente às outras em termos de preço, capacidade, velocidade e espaço físico ocupado? **(2 Val.)**

## Parte Prática

1. Realize um programa em *Assembly*, que permita calcular e guardar as médias semanais de um conjunto de medições de consumo de combustível. Assuma a existência de um vector, *VectorIni*, onde estão representados os consumos diários referentes a várias semanas, sendo o final do vector representado pelo valor -1. Considere que o registo de consumo foi feito apenas uma vez por dia, que uma semana tem 7 dias (registados sequencialmente) e que o vector *VectorIni* possui registos de semanas completas. Todos os consumos estão expressos em decilitros, por exemplo, um consumo referente a 2,3 litros está armazenado como 23 decilitros. Assuma que não há consumos superiores a 25,0 litros. Os resultados deverão ser armazenados noutro vector, *VectorFin*, em que cada elemento conterá uma média semanal. As médias também serão representadas em decilitros e serão arredondadas ao inteiro mais próximo. **(2,5 Val.)**

2. Elabore um programa em *Assembly* que desenhe no écran (usando acesso directo à memória de vídeo) um triângulo retângulo preenchido com determinado carácter, em que a dimensão da base é igual à altura. Para isso, deve ser declarado no programa a dimensão da base e o carácter utilizado para desenhar o triângulo. O Triângulo deve começar a ser desenhado numa posição do ecrã definido por LINHA (0..24) e COLUNA (0..79) também declarados no segmento de dados. Não é necessário validar os limites do monitor.

Exemplo, caso a dimensão seja 5 e o carácter x, será exibido o seguinte triângulo:

```
x
xx
xxx
xxxx
xxxxx
```

**(2,5 Val.)**

Nota: A memória de vídeo, no caso de sistemas policromáticos, tem início na localização B800h:0000h.

```
.8086
.model small
.stack 2048

DATA_HERE    SEGMENT
...
DATA_HERE    ENDS
CODE_HERE    SEGMENT
    ASSUME CS:CODE_HERE, DS:DATA_HERE
START:mov     ax, DATA_HERE
        mov     ds, ax
        mov     bx, 0b800h
        mov     es, bx
        ...
        mov     ah, 4ch
        int     21h
CODE_HERE    ENDS
END START
```