



## ARQUITETURA DE COMPUTADORES | 21010 | 2023/2024

Pretende-se implementar no P3 um programa que considere uma carta com 3 tipos: operador; número; letra. Cada tipo tem 4 possibilidades de acordo com a seguinte tabela:

| Tipos    | Formas  | Códigos |
|----------|---------|---------|
| Operador | + - * / | 0 1 2 3 |
| Número   | 1 2 3 4 | 0 1 2 3 |
| Letra    | A B C D | 0 1 2 3 |

Uma carta é codificada num só número de 6 bits, sendo os dois bits menos significativos para letra, os seguintes para o número, e os mais significativos para o operador. Assim, a carta \*2D é convertida nos códigos (2,1,3), que em binário fica (10b,01b,11b), e na codificação de 6 bits fica 100111b ou 39 em decimal.

- a) [1] Pretende-se que implemente uma sub-rotina para descodificar uma carta nos códigos de cada tipo.
- **Entrada:** R1 tem o número com a carta codificada no código de 6 bits
  - **Saída:**
    - i. R2 - código do operador
    - ii. R3 - código do número
    - iii. R4 - código da letra
- b) [1] Pretende-se que implemente uma sub-rotina que receba uma lista de códigos de cartas. A lista termina com um código inválido, ou seja, um valor igual ou superior a 64. Deverá ser calculada a frequência de cada código.
- **Entrada:** R1 tem o endereço da lista de códigos das cartas
  - **Saída:**
    - i. FreqOperador - endereço de 4 posições, com a frequência de cada código para este operador;
    - ii. FreqNumero - endereço de 4 posições;
    - iii. FreqLetra - endereço de 4 posições.

- c) [1] Pretende-se que implemente uma sub-rotina para verificar se duas cartas têm o mesmo código num dos tipos, operador, número ou letra.

- **Entrada:**
  - R1 - código da carta 1
  - R2 - código da carta 2
- **Saída:** R3 - 1 se um dos códigos é igual, 0 c.c.

- d) [1] Pretende-se que implemente uma sub-rotina para saber quantos pares de cartas têm um tipo igual numa lista de código de cartas.

- **Entrada:** R1 tem o endereço da lista de códigos das cartas
- **Saída:** R1 - resultado

Execute os testes seguintes às sub-rotinas implementadas. Apresentamos já alguns resultados, para validação.

Testes A:

| Entrada - R1 | Saída - R2, R3, R4 |
|--------------|--------------------|
| 23           | 1,1,3              |
| 57           | 3,2,1              |

Testes B:

| Entrada - R1                                   | Saída - FreqOperador, FreqNumero, FreqLetra |
|--|---|
| 12,48,35,16,3,14,45,11,47,34,15,2,13,44,20,100 | FreqOperador - 7, 2, 5, 1                   |

Testes C:

| Entrada - R1, R2 | Saída - R3 |
|------------------|------------|
| 13, 40           | 0          |
| 40, 31           | 0          |
| 13, 31           | 1          |
| 55, 22           |            |

Testes D:

| Entrada - R1   | Saída - R1 |
|--|------------|
| 12,48,35,16,3,14,45,11,47,34,15,2,13,44,20,100   |            |
| 13,40,31,100   | 1          |
| 41,40,63,16,58,54,39,23,3,53,2,16,39,6,62,39,1,17,21,6,17,0,24,57,43,51,31,100                 |            |
| 41,56,52,17,12,22,4,8,3,6,57,60,9,37,55,27,62,4,26,20,42,52,11,0,28,6,38,18,10,16,36,61,36,100 |            |

De seguida apresenta-se um programa de teste que deve utilizar de modo a executar os testes após completar com as suas alíneas. Caso não tenha realizado uma alínea, comente o código que chamaria essa alínea.

## Programa de teste:

```

; zona de dados
ORIG 8000h

ResA      TAB 16      ; 8000h, 8008h
FreqOperador STR 0,0,0,0 ; 8010h,
FreqNumero STR 0,0,0,0 ; 8014h,
FreqLetra STR 0,0,0,0 ; 8018h,
ResC      TAB 4      ; 801ch
ResD      TAB 8      ; 8020h

; cartas
Lista1 STR 12,48,35,16,3,14,45,11,47,34,15,2,13,44,20,100
Lista2 STR 13,40,31,100
Lista3 STR 41,40,63,16,58,54,39,23,3,53,2,16,39,6,62,39,1,17,21,6,17,0,24,57,43,51,31,100
Lista4 STR 41,56,52,17,12,22,4,8,3,6,57,60,9,37,55,27,62,4,26,20,42,52,11,0,28,6,38,18,10,16,36,61,36,100

; zona do código
ORIG 0000h

Início: ; inicialização do stack
        MOV R1, fdifh
        MOV SP, R1

; teste alínea A
        MOV R1, 23
        ;CALL ---- identificador da subrotina solicitada na alínea A
        MOV R1, ResA
        MOV M[R1], R2
        MOV M[R1+1], R3
        MOV M[R1+2], R4

        MOV R1, 57
        ;CALL ---- identificador da subrotina solicitada na alínea A
        MOV R1, ResA
        MOV M[R1+8], R2
        MOV M[R1+9], R3
        MOV M[R1+10], R4

; teste alínea B
        MOV R1, Lista1
        ;CALL ---- identificador da subrotina solicitada na alínea B

; teste alínea C
        MOV R1, 13
        MOV R2, 40
        ;CALL ---- identificador da subrotina solicitada na alínea C
        MOV R1, ResC
        MOV M[R1], R3

        MOV R1, 40
        MOV R2, 31
        ;CALL ---- identificador da subrotina solicitada na alínea C
        MOV R1, ResC
        MOV M[R1+1], R3

        MOV R1, 13
        MOV R2, 31
        ;CALL ---- identificador da subrotina solicitada na alínea C
        MOV R1, ResC
        MOV M[R1+2], R3

        MOV R1, 55
        MOV R2, 22
        ;CALL ---- identificador da subrotina solicitada na alínea C
        MOV R1, ResC
        MOV M[R1+3], R3

; teste alínea D
        MOV R1, Lista1
        ;CALL ---- identificador da subrotina solicitada na alínea D
        MOV R2, ResD
        MOV M[R2], R1

        MOV R1, Lista2
        ;CALL ---- identificador da subrotina solicitada na alínea D
        MOV R2, ResD
        MOV M[R2+1], R1

        MOV R1, Lista3
        ;CALL ---- identificador da subrotina solicitada na alínea D
        MOV R2, ResD
        MOV M[R2+2], R1

        MOV R1, Lista4
        ;CALL ---- identificador da subrotina solicitada na alínea D
        MOV R2, ResD
        MOV M[R2+3], R1

Fim:    BR Fim
```

Deve apresentar o resultado da execução do programa de teste, os ciclos de relógio e instruções, bem como a zona da memória em que são gravados os resultados. A execução do programa de teste sem o código de qualquer alínea realizada:

```
Clock: 446
Instructions: 42
8000 : 0000 0000 0000 0000 0000 0000 0000 0000 .....
8008 : 0000 0000 0000 0000 0000 0000 0000 0000 .....
8010 : 0000 0000 0000 0000 0000 0000 0000 0000 .....
8018 : 0000 0000 0000 0000 0000 0000 0000 0000 .....
8020 : 8028 8038 803c 8058 0000 0000 0000 0000 .....
```

**BOM TRABALHO!**

### **Avaliação**

#### **Cotação:**

A cotação encontra-se junto de cada uma das alíneas, entre [].

#### **Critérios de Correção:**

Funcionalidade: 50%

Simplicidade e Modularidade: 10%

Eficiência (serão contabilizados o número de instruções e ciclos de relógio): 10%

Apresentação do código (indentação e comentários): 10%

Relatório (Legibilidade e Justificação dos Resultados e das Opções): 20%

#### **Descontos:**

Trabalhos entregues que não estejam em conformidade com as regras de entrega do e-fólio B: até 10%

Código sem comentários, ou apenas com comentários a refletir o significado da instrução (exemplo MOV R1,R2 ;mover o conteúdo de R2 para R1) : até 50%

Deteção de fraude (total ou parcial): 100%

Trabalhos entregues após a data limite (máximo 24h) : 10%

### **Regras para entrega do e-fólio B:**

#### **Forma de entrega:**

Deverá ser entregue um relatório em formato pdf ou Word até 5 páginas A4, com todos os cálculos e todas as opções tomadas na construção dos programas. Em anexo deve colocar todo o código e resultados obtidos (número de instruções, ciclos de relógio e conteúdo da memória de 8000h a 8027h). O código deve estar num formato que permita a seleção de modo a ser copiado e colado para o simulador do P3.

Não são aceites entregas fora da plataforma Moodle.