

Arquitetura de Computadores

2007/2008 2º Semestre

1º Teste (A) - 30/04/2008

Número: _____ Nome: _____

INSTRUÇÕES:

- A duração da prova é de 1,5 horas.
- Responda apenas nesta “Folha de Respostas”; nada mais será recebido.
- Identifique esta folha com o seu número e nome de forma bem legível.
- A cotação das perguntas está indicada entre parênteses à direita do texto.
- Nas perguntas Verdade/Falso ou de escolha múltipla as respostas erradas descontam.

Folha de Respostas

Grupo I

	R1	R2	R3	R4	R5	R6	R7	PC	SP	RE	
1.1 a)	0016h	F3A0h						011Ah		0000h	(1)
1.1 b)			9642h					011Ah		0002h	(1)
1.1 c)								0210h			(1)

1.2 a)

443Eh	0003h
-------	-------

 (1) (443Eh <-> 0100 01 0000 11 1110 b)

1.2 b)

000Dh	000Eh
-------	-------

 (1)

1.2 c)

A1F3h

 (1,5)

1.2 d) (1,5)

	Endereço	Dados	L / E
1	000Dh	443Eh	L
2	000Eh	0003h	L
3	A1F6h	0000h	L
4	A1F6h	0001h	E
5			

Grupo II

(A-F)
2.1 a)

B

 (2)
2.1 b)

C

 (1,5)
2.1 c)

A

 (1)

2.2 a)

Instruções
MOV R1, R6
MOV R6, R5
MOV R5, R1

 (1)

(R1 ou R2, R3 ou R4)

Ou, como não são relevantes os bits de estado (flags)

naquele troço de programa: XOR R5, R6
XOR R6, R5
XOR R5, R6

V.S.F.F.

- (A-F)
- 2.2 b)

D

 (1) NOTA sobre 2.2 b): A resposta efectivamente correcta é “min(A,B) -1”.
- 2.2 c)

A

 (1) Por essa razão também foi considerada certa a resposta F.

Grupo III

- 3.1 a)

MOV R1, FDFFh
MOV SP, R1

 (0,5) (FDFFh ou outro valor adequado – endereço elevado que esteja disponível; R1 ou outro registo)

- 3.1 b)

8021h

 (0,5)

- 3.2 a)

ORIG FE05h
INT5 WORD Bot5

 (1) Ou, por exemplo, MOV R1, Bot5
MOV M[FE05h], R1

- 3.2 b)

RE
0000h

 (0,5)

- 3.2 c)

E	Z	C	N	O
1	?	?	?	1

 (1)

- 3.3 (A-E)
- | |
|---|
| C |
|---|

 (1)

Critério de classificação do Grupo I**1. a), 1. b) e 1. c)**

Legenda:

C = Número de respostas com posição e valor certos

S = Número de respostas com posição certa, valor errado

E = Número de respostas com posição e valor errados

N = Número total de respostas correctas

$$Nota[0,1] = \frac{(C + \frac{S}{2})}{MAX\{N; (C + S + E)\}}$$

I

1.1 Considere os seguintes valores para os registos do processador P3:

R1	R2	R3	R4	R5	R6	R7	PC	SP	RE
B79Dh	0020h	69BEh	0040h	4C75h	7DF5h	A813h	0119h	FA5Dh	0006h

Para as alíneas a), b) e c), indique na folha de respostas qual o novo valor, em **hexadecimal**, para todos (**e apenas**) os registos que são alterados pelo ciclo completo de execução de cada instrução. Use o símbolo ? caso não tenha informação suficiente para determinar o novo valor de um registo.

As alíneas são independentes, isto é, assuma como valores iniciais para cada alínea os indicados na tabela acima.

1.1 a) MUL R1, R2 (1)

1.1 b) NEG R3 (1)

1.1 c) JMP.C 0210h (1)

1.2 Considere o seguinte troço de código em Assembly do P3:

```

                                ORIG    0000h
                                MOV     R1, A1FFh
                                MOV     SP, R1

                                PUSH    R0
                                PUSH    3
                                CALL    factorial
                                POP     R7
fim:                            BR      fim

factorial: CMP     M[SP+2], R0
                                BR.NZ  chamaDecN
                                INC     M[SP+3]
                                RETN    1
chamaDecN: MOV     R1, M[SP+2]
                                DEC     R1
                                PUSH    R0
                                PUSH    R1
                                CALL    factorial
                                POP     R1
                                MOV     R2, M[SP+2]
                                MUL     R1, R2
                                MOV     M[SP+3], R2
                                RETN    1

```

Relativamente à instrução **INC M[SP+3]** do código acima apresentado, indique na folha de respostas, utilizando a base de representação hexadecimal:

1.2 a) A respectiva codificação (código objecto); (1)

1.2 b) O(s) endereço(s) de memória onde a referida instrução se encontra armazenada; (1)

1.2 c) O valor do stack-pointer (SP) aquando da execução da instrução (sugestão: represente o estado da pilha ao longo da execução do programa, até ao instante em que a instrução é executada). (1,5)

1.2 d) A sequência de acessos à **memória** durante o ciclo completo de execução da instrução, especificando o endereço, dados e tipo de acesso (**Leitura/Escrita**). Utilize a tabela presente na folha de respostas. **(1,5)**

Nota 1: A tabela tem 5 linhas, utilize apenas as que considerar necessárias.

Nota 2: Caso não tenha respondido às perguntas anteriores, utilize os símbolos **OP**, **ADDR** e **SP** para indicar a respectiva codificação, endereço e valor do stack-pointer (penalização de 50%).

II

2.1 Vamos avaliar as características de geração de código de 4 compiladores de C para o P3. Para tal compilou-se o programa seguinte tendo sido produzidos os resultados indicados na tabela.

```
#define SIZE 64
#define K 27
register short int i;
short int vec[SIZE];

for (i=0; i<SIZE; i=i+1) {
    vec[i] = i+K;
}
```

Considere que não há *overflows* ou *underflows* nas operações aritméticas.

A: 1 SIZE EQU 64 2 K EQU 27 3 VEC TAB SIZE 4 MOV R3,R0 5 LOOP: CMP R3,SIZE 6 BR.NN NEXT 7 MOV R7,K 8 ADD R7,R3 9 MOV M[R3+VEC],R7 10 INC R3 11 BR LOOP 12 NEXT:	B: 1 SIZE EQU 64 2 K EQU 27 3 VEC TAB SIZE 4 MOV R3,0 5 LOOP: CMP R3,SIZE 6 BR.NN NEXT 7 MOV R7,K 8 ADD R7,R3 9 MOV M[R3+VEC],R7 10 INC R3 11 JMP LOOP 12 NEXT:
C: 1 SIZE EQU 64 2 K EQU 27 3 VEC TAB SIZE 4 MOV R7,K 5 MOV R6,SIZE 6 MOV R3,R0 7 LOOP: CMP R3,R6 8 BR.NN NEXT 9 MOV M[R3+VEC],R7 10 INC R7 11 INC R3 12 BR LOOP 13 NEXT:	D: 1 SIZE EQU 64 2 K EQU 27 3 VEC TAB SIZE 4 MOV R3,R0 5 LOOP: CMP R3,SIZE 6 BR.NN NEXT 7 MOV M[R3+VEC],R3+K 8 INC R3 9 BR LOOP 10 NEXT:

2.1 a) Indique, na folha de respostas, que programas produzem um resultado correcto (na geração de código e no algoritmo). (2)

- | |
|--|
| A: Programas A e C.
B: Programas A, B e C.
C: Programas C e D.
D: Programas A, C e D.
E: Todos os programas são correctos.
F: Nenhuma das respostas anteriores está correcta. |
|--|

2.1 b) Dos programas correctos indique qual é o mais rápido (1,5)

A: Programa A.	B: Programa B.	C: Programa C.	D: Programa D.
-----------------------	-----------------------	-----------------------	-----------------------

2.1 c) Dos programas correctos indique qual é o mais compacto (o que ocupa menos memória). (1)

A: Programa A.	B: Programa B.	C: Programa C.	D: Programa D.
-----------------------	-----------------------	-----------------------	-----------------------

2.2 Considere agora uma versão muito simples da arquitectura P3 – pico-P3. O pico-P3 executa um sub-conjunto das instruções do P3: Não tem instruções MUL e DIV; as instruções aritméticas e lógicas só operam com registos; só os MOV acedem a variáveis na memória – MOV op1,op2, em que op = {Cte. | Reg. | M[Reg.]}. No restante o pico-P3 é compatível com o P3.

Considere o programa seguinte do pico-P3. O programa tem como operandos A = M[R1] e B = M[R2]; o resultado é guardado em B.

```

1  A:  MOV    R5,M[R1]
2      MOV    R6,M[R2]
3      CMP    R5,R6
4      BR.N   M
5      XCH    R5,R6
6  M:  CMP    R5,0
7      BR.NZ  NZ
8  Z:  MOV    R6,0
9      JMP    E
10 NZ:  DEC    R5
11      CMP    R5,0
12      BR.Z   E
13      MOV    R7,R6
14 S:  ADD    R6,R7
15      DEC    R5
16      BR.P   S
17 E:  MOV    M[R2],R6
18

```

2.2 a) Se o pico-P3 também não tivesse instrução XCH indique, na folha de respostas, que sequência de instruções substituiria a instrução na linha 5. (Use o menor número de instruções que considere necessário) (1)

2.2 b) Quantas iterações executa o programa? (1)

A: A iterações.	B: B iterações.	C: max (A,B) iterações.	D: min (A,B) iterações.	E: (A+B) iterações.	F: Nenhuma das anteriores.
------------------------	------------------------	--------------------------------	--------------------------------	----------------------------	-----------------------------------

2.2 c) Qual é a função executada pelo programa? (1)

A: $B \leftarrow A \times B$	B: $B \leftarrow (A+B)^2$	C: $B \leftarrow (A-B)^2$	D: $B \leftarrow A-B ^2$	E: $B \leftarrow A^2 + B^2$	F: Nenhuma das anteriores.
-------------------------------------	----------------------------------	----------------------------------	----------------------------------	------------------------------------	-----------------------------------

III

3.1 Considere um programa Assembly para o P3.

a) Indique, na folha de respostas, as instruções que usaria para inicializar o stack-pointer (SP) caso esse programa utilize três rotinas de tratamento de interrupção. (0,5)

b) Assuma que as rotinas de tratamento de interrupção estão associadas aos botões de pressão I0, I5 e ao temporizador. Indique na folha de respostas, em hexadecimal, qual o valor da máscara de interrupções a usar para permitir essas interrupções. (0,5)

3.2 Considere que foi desenvolvida a seguinte rotina para tratar a interrupção gerada pelo botão de pressão I5:

```

1 Bot5:  ORIG      0200h
2         PUSH     R1
3         MOV      R1, R0
4         BR.Z     L1
5         INC      M[CounterX]
6 L1:     INC      R1
7         OR       M[SP+3], R1
8         POP      R1
9         RTI

```

a) Indique na folha de respostas as instruções/pseudo-instruções que usaria para preencher correctamente a tabela de vectores de interrupção relativamente à rotina acima. (1)

b) Indique o valor do Registo de Estado (RE), em hexadecimal, quando se atinge a linha 3 da rotina acima. Se considerar não ser possível conhecer o valor de RE indique '?'. (0,5)

c) Indique o valor individual dos bits de estado (*flags*) imediatamente após a execução da instrução RTI. Responda usando 0, 1 ou ? se não for possível conhecer o valor de um bit. (1)

3.3 Suponha que durante a execução da rotina Bot5 apresentada acima ocorre um pedido de interrupção gerado pelo temporizador. Indique, na folha de respostas, qual das seguintes afirmações está correcta. (1)

- A - A rotina de tratamento da interrupção do temporizador começará a ser executada imediatamente após conclusão da instrução que esteja em curso.
- B - O pedido de interrupção é perdido.
- C - A rotina de tratamento da interrupção será executada imediatamente a seguir à conclusão da rotina Bot5.
- D - A rotina de tratamento da interrupção só será executada após ser encontrado um ENI no programa.
- E - Nenhuma das opções anteriores.