

Nome Aluno:

Nº Aluno:

Deve assinalar a resposta correta, circundando-a nas perguntas de escolha múltipla e inserindo um "X" nas restantes. Se precisar de alterar alguma resposta deve riscá-la e indicar a nova resposta. As respostas incorretas, serão penalizadas.

**Duração Total do Teste:** 1h:30m

**Sem consulta**

24 de Maio de 2014

1. Das seguintes instruções indique quais são possíveis:

a. MOV CL,BL	b. MOV CL,[BL]	c. MOV CL,[BX]	d. MOV CL,BX
e. MOV CX,BL	f. MOV CX,[BL]	g. MOV CX,[BX]	h. MOV CX,BX

2. Considerando os seguintes dados:

AX = 1234H

BX = 5678H

CX = 9ABCH

DX = DEF8H

**Data Segment:**

ENDEREÇO	CONTEÚDO
1122H	0CH
1123H	20H
1124H	BDH
5677H	3FH
5678H	DDH
5679H	50H

**Stack Segment:**

ENDEREÇO	CONTEÚDO
1122H	1DH
1123H	31H
1124H	CEH
5677H	40H
5678H	EEH
5679H	61H

O **resultado final** do seguinte conjunto de instruções tem o seguinte efeito:

MOV BX,1122H

MOV AL,[BX]

SUB CL,AL

INC BX

MOV [BX],CL

Indique a veracidade de cada uma das instruções:

V	F	
		a) BH fica com o valor 11H.
		b) BL fica com o valor 22H.
		c) BX fica com o valor 1123H.
		d) BX fica com o valor 3322H.
		e) AL fica o valor 22H.
		f) AL fica com o valor 1DH.
		g) AL fica com o valor 0CH.
		h) CL fica com o valor 0CH.
		i) CL fica com o valor B0H.
		j) O conteúdo do endereço SS:1123H fica com o valor B0H.
		k) O conteúdo do endereço DS:1123H fica com o valor B0H.
		l) BX fica com o valor 200CH.
		m) DS:1122H fica com o valor 20H.
		n) DS:1123H fica com o valor B0H.
		o) DS:1123H fica com o valor 0CH.

3. Dado o seguinte programa, indique a veracidade de cada uma das afirmações:

```

1: . 8086
2: . model small
3: . stack 2048

4: dados segment para 'data'
5:     vector1 db 1H, 2H, 3H, 4H, 5H, 6H, 7H, 8H, 9H, 10H
6:     vector2 db 11, 12, 13, 14, 15, 20, 17, 18, 19, 16
7:     Num db 10
8: dados ends

9: codigo segment para 'code'
10:     main proc
11:         assume cs:codigo, ds:dados
12:         mov ax, dados
13:         mov ds, ax

14:         mov cl, Num
15:         mov ch, 0
16:         mov si, 0
17:     s1: mov al, vector1[si]
18:         mov ah, vector2[si]
19:         mov vector2[si], al
20:         mov vector1[si], ah
21:         inc si
22:         cmp cx, si
23:         jne s1

24:         mov ah, 4Ch
25:         int 21h

26:     main endp
27:     codigo ends
28: end main

```

V	F	
		a) Vector1[9] tem a mesma informação que Vector2[9].
		b) Vector1 armazena 10 bytes.
		c) Vector1 armazena 10 double bytes.
		d) Assumindo que ao segmento de dados foi atribuído o valor 1234H, a variável Num fica alojada em 1234H:0020H.
		e) No final da execução do programa o registo CX tem o valor 10.
		f) O resultado do programa seria o mesmo se se trocassem as instruções das linhas 18 e 19 entre si.
		g) A instrução da linha 15 é desnecessária no contexto deste programa.
		h) As instruções das linhas 14 e 15 servem para colocar o registo CX com o valor de Num.
		i) Da 1ª vez que é executada, a instrução da linha 21 faz com que SI fique com o valor 1.
		j) A instrução da linha 22 tem como objetivo verificar se os elementos dos 2 vetores são iguais.
		k) A instrução da linha 22 tem como objetivo verificar se se chegou ao último elemento dos vetores.
		l) As instruções entre as linhas 17 e 23 são executadas enquanto SI tiver um valor diferente de 10.
		m) A instrução da linha 24 é executada quando SI tiver o valor 11.
		n) A instrução da linha 24 é executada quando CX tiver o valor 11.
		o) O programa tem como objetivo comparar os elementos de dois vetores.

4. Sabendo que pretende somar o valor presente em CL ao valor em AH, deixando o resultado em AX, indique qual/quais o(s) grupo(s) de instruções que o permite realizar corretamente:

a. MOV AL,0 ADD AH,CL ADC AH,0	b. MOV CH,0 ADD AH,CL ADC AH,0
c. MOV AL,0 MOV CH,0 ADD AX,CX	d. MOV AX,0 MOV AL,AH MOV CH,0 ADD AX,CX
e. MOV CH,0 ADD AH,CL ADC AL,0	f. MOV CH,0 MOV AL,AH ADD AL,CL ADC CH,0
g. MOV AH,0 MOV AL,AH MOV CH,0 ADD AX,CX	h. MOV CH,0 MOV AL,AH MOV AH,0 ADD AX,CX

5. Sabendo que pretende dividir o valor 50 pelo valor 4 que se encontra no registo AL, indique qual/quais o(s) grupo(s) de instruções que o permite realizar corretamente:

a. MOV BX,0 MOV DX,0 MOV BL,AL MOV AX,50 DIV BX	b. MOV AX,50 MOV BL,4 MOV BH,0 DIV BX	c. MOV BX,50 DIV AL
d. MOV AX,50 DIV 4	e. MOV DX,0 MOV AL,50 MOV AH,0 DIV 4	f. MOV BL,AL MOV AX,50 DIV BL

**DIV OPER**

- Se OPER for de 8 bits divide AX por OPER, ficando o quociente em AL e o resto em AH.

- Se OPER for de 16 bits divide DX:AX por OPER, ficando o quociente em AX e o resto em DX.