

## Examen 2

Assignatura	Codi	Data	Hora inici
Estructura de computadores	05.573		

05.573 17 01 15 EX

Enganxeu en aquest espai una etiqueta identificativa  
amb el vostre codi personal  
Examen

### Fitxa tècnica de l'examen

- Comprova que el codi i el nom de l'assignatura corresponen a l'assignatura en la qual estàs matriculat.
- Només has d'enganxar una etiqueta d'estudiant a l'espai corresponent d'aquest full.
- No es poden adjuntar fulls addicionals.
- No es pot realitzar la prova en llapis ni en retolador gruixut.
- Temps total: 2 h.
- En cas que els estudiants puguin consultar algun material durant l'examen, quin o quins materials poden consultar?  
No es pot utilitzar calculadora, ni material auxiliar.
- Valor de cada pregunta: Pregunta 1 (20%); Pregunta 2 (40%); Pregunta 3 (40%)
- En cas que hi hagi preguntes tipus test: Descompten les respostes errònies? NO Quant?
- Indicacions específiques per a la realització d'aquest examen:

### Enunciats

No es pot utilitzar calculadora. Cal saber interpretar un valor en binari, decimal o hexadecimal per a realitzar l'operació que es demani. I el resultat s'ha d'expressar en el format corresponent.

## Examen 2

Assignatura	Codi	Data	Hora inici
Estructura de computadores	05.573		

### Valoració de les preguntes de l'examen

#### Pregunta 1 (20%)

Pregunta sobre la pràctica.

**Cal completar les instruccions marcades o afegir el codi que es demana.**

**Els punts suspensius indiquen que hi ha més codi però no l'heu de completar.**

NOTA: En cas que el codi proposat en cada pregunta no es correspongui amb la forma que vosaltres plantejaríeu la resposta, podeu reescriure el codi o part del codi segons el vostre plantejament.

**1.1: 10%**

**1.2: 10%**

#### Pregunta 2 (40%)

**2.1: 15%**

**2.2: 15%**

**2.3: 10%**

#### Pregunta 3 (40%)

**3.1: 15%**

**3.1.1: 15%**

**3.1.2: 10%**

**3.2: 25%**

**3.2.1: 15%**

**3.2.2: 10%**

## Examen 2

Assignatura	Codi	Data	Hora inici
Estructura de computadors	05.573		

### Pregunta 1

#### 1.1 Pràctica – Part obligatòria

Escriure un fragment de codi assemblador de la subrutina GetPlay que tregui per pantalla els dígit introduïts al vector play. Supposeu que ja els heu entrat per teclat.

```

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
; GetPlay
; Llegir una jugada.
; Primer netejar l'espai on es llegeix la combinació amb espais en
; blanc, cridant la subrutina clearArea
; Inicialitzar a zeros el vector play.
; Posar showChar=1, per a indicar que Printch mostri els
; caràcters llegits.
; Llegir la jugada (5 dígit) i emmagatzemar-la a play
; Imprimir, cridant a la subrutina printch, la combinació llegida de teclat
; Printch treu per pantalla el caràcter que hi ha al registre AL
;
; Variables utilitzades:
; play      : vector que emmagatzema la jugada
; showChar: 1: mostrar el caràcter llegit
; carac    : variable on la subrutina getch deixa el caràcter
; pitjat
;
; Paràmetres d'entrada :
; Cap
;
; Paràmetres de sortida:
; Cap
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
GetPlay:

```

## Examen 2

Assignatura	Codi	Data	Hora inici
Estructura de computadors	05.573		

### 1.2 Pràctica – Part opcional

Completar el codi de la subrutina PrintTries per a que posicioni el cursor en la posició (3,60) de la pantalla

```

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;PrintTries
; Mostrar els intents que queden
; Situar el cursor a la fila 3, columna 60 cridant la subrutina
; gotoxy, passant fila i columna mitjançant les variables row i
; col. És necessari guardar el valor actual de row per a poder restaurar-lo
; abans de sortir de la subrutina.
; Mostra el valor de la variable tries cridant a la subrutina
; printch, passant el caràcter associat als intents com a
; paràmetre a través del registre al.
; Per obtenir el caràcter associat als intents, codi ASCII del
; número, cal sumar al valor numèric dels intents, tries, el valor
; decimal 48.
;
; Variables utilitzades:
; tries: nombre d'intents que queden
;
; Paràmetres d'entrada : Cap
;
; Paràmetres de sortida: Cap
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
PrintTries:
    ...
    push rdi
    push rsi
    push rax

    mov _____, [row]
    mov dword[row], _____
    mov _____, 60
    mov edi, [row]
    mov esi, [col]
    call _____
    mov [row], eax

    mov al, [tries]
    add al, 48
    mov dil, al
    call printch

    pop rax
    pop rsi
    pop rdi
    pop rax

    ...
    ret

```

## Examen 2

Assignatura	Codi	Data	Hora inici
Estructura de computadores	05.573		

### Pregunta 2

#### 2.1

L'estat inicial del computador CISCA just abans de començar l'execució de cada fragment de codi (en cada apartat) és el següent:

R0 = 00000A10h R1 = 00000B20h R2 = 00000C30h	M(00000A10h) = 0000F00Fh M(00000B20h) = 0000F000h M(00000C30h) = 0000FF00h M(000000F0h) = 00000001h M(000FF0A0h) = 0000000Ah	Z = 0, C = 0, S = 0, V = 0
--	--	----------------------------

Quin serà l'estat del computador després d'executar cada fragment de codi? (només modificacions, excloent-hi el PC).

a)	<pre> XOR  R0, R0 JE    END ADD   R0, [R2] END:  MOV  R1, [R2] </pre>
Z = , S = , C = , V =	

b)	<pre> DEC  [000000F0h] SUB  [000000F0h], R2 </pre>
Z = , S = , C = , V =	

## Examen 2

Assignatura	Codi	Data	Hora inici
Estructura de computadors	05.573		

### 2.2

Donat el següent codi en alt nivell:

```
i = MAX;
while (i>0) { A[i] = 0; i = i - 1; }
```

A és un vector de 8 elements de 4 bytes cadascun. Es proposa la següent traducció a CISCA on hem deixat 5 llocs per omplir:

```

PLUS:      MOV  _____,  MAX
            CMP  R0,  0
            _____
            END
            MOV  R1,  R0
            _____
            R1,  4
            MOV  _____,  0
            DEC  R0
            JMP  PLUS
END:       MOV  _____,  0
```

## Examen 2

Assignatura	Codi	Data	Hora inici
Estructura de computadors	05.573		

### 2.3

Donat el següent fragment de codi d'un programa en llenguatge ensamblador de CISCA:

```
SUB  R3, 10
MOV  [R1], R3
ADD  R1, 4
```

Tradueu-lo a llenguatge màquina i expresseu-lo en la següent taula. Supposeu que la primera instrucció del codi s'assembla a partir de l'adreça 00000D10h (que és el valor del PC abans de començar l'execució del fragment de codi). En la següent taula useu una fila per a codificar cada instrucció. Si suposem que la instrucció comença en l'adreça @, el valor Bk de cadascun dels bytes de la instrucció amb adreces @+k per a k=0, 1,... s'ha d'indicar en la taula en hexadecimal en la columna corresponent (recordeu que els camps que codifiquen un desplaçament en 2 bytes o un immediat o una adreça en 4 bytes ho fan en format little endian, això cal tenir-lo en compte escrivint els bytes de menor pes, d'adreça més petita, a l'esquerra i els de major pes, adreça major, a la dreta). Completeu també la columna @ que indica per a cada fila l'adreça de memòria del byte B0 de la instrucció que es codifica en aquesta fila de la taula.

A continuació us donem com ajuda les taules de codis:

Taula de codis d'instrucció

B0	Instrucció
10h	MOV
21h	SUB
20h	ADD

Taula de modes d'adreçament (Bk<7..4>)

Camp mode Bk<7..4>	Mode
0h	Immediat
1h	Registre
2h	Memòria
3h	Indirecte
4h	Relatiu
5h	Indexat
6h	Relatiu a PC

Taula de modes d'adreçament (Bk<3..0>)

Camp mode Bk<3..0>	Significat
Nº registre	Si el mode ha d'especificar un registre
0	No s'especifica registre.

		Bk per a k=0..10											
@	Assemblador	0	1	2	3	4	5	6	7	8	9	10	
	SUB R3, 10												
	MOV [R1], R3												
	ADD R1,4												

## Examen 2

Assignatura	Codi	Data	Hora inici
Estructura de computadors	05.573		

### Pregunta 3

#### 3.1. Memòria cau

##### Memòria cau completament associativa (FIFO)

Tenim un sistema de memòria en el que tots els accessos es fan a paraula (no ens importa quina és la mida d'una paraula). Suposarem que l'espai d'adreces de memòria es descompon en blocs de 8 paraules. Cada bloc comença en una adreça múltiple de 8. Així, el bloc 0 conté les adreces 0, 1, 2, 3, 4, 5, 6, 7, el bloc 1, les adreces 8, 9, 10, 11, 12, 13, 14, 15, i el bloc N les adreces  $8*N$ ,  $8*N+1$ ,  $8*N+2$ ,  $8*N+3$ ,  $8*N+4$ ,  $8*N+5$ ,  $8*N+6$ ,  $8*N+7$ .

Suposem que el sistema també disposa d'una memòria cau de 4 línies (on cada línia té la mida d'un bloc, es a dir, 8 paraules). Aquestes línies s'identifiquen com a línies 0, 1, 2 i 3. Quan es fa referència a una adreça de memòria principal, si aquesta adreça no es troba a la memòria cau, es porta tot el bloc corresponent des de la memòria principal a una línia de la memòria cau (així si fem referència a l'adreça 2 de memòria principal portarem el bloc format per les paraules 0, 1, 2, 3, 4, 5, 6, 7).

Suposem que el sistema fa servir una política d'emplaçament completament associativa, de manera que qualsevol bloc de la memòria principal es pot portar a qualsevol bloc de la memòria cau. Si trobem que la cau ja està plena, es fa servir un **algorisme de reemplaçament FIFO**.

L'execució d'un programa genera la següent llista de lectures a memòria:

7, 8, 24, 23, 38, 39, 40, 45, 46, 47, 48, 16, 18, 20, 32, 40, 48, 33, 41, 49

Inicialment la memòria cau és buida i s'omple seqüencialment començant per la línia 0.

**3.1.1** Completar la taula per a mostrar l'evolució de la cau durant l'execució del programa. Per a cada accés cal omplir una columna indicant si es tracta d'un encert o una fallada.

Si és un encert escriurem E en la línia corresponent davant de les adreces del bloc, si és una fallada escriurem F i s'indicarà el nou bloc que es porta a la memòria cau en la línia que li correspongui, expressat de la forma  $b(a_0 - a_7)$  on  $b$ : número de bloc, i  $(a_0 - a_7)$  són les adreces del bloc, on  $a_0$  és la primera adreça del bloc i  $a_7$  és la vuitena (darrera) adreça del bloc.

Línia	Estat Inicial	7	8	24	23	38
0	----	F 0 (0 - 7)				
1	----					
2	----					
3	----					

Línia	39	40	45	46	47	48
0						
1						
2						
3						



## Examen 2

Assignatura	Codi	Data	Hora inici
Estructura de computadors	05.573		

Línia	16	18	20	32	40	48
0						
1						
2						
3						

Línia	33	41	49			
0						
1						
2						
3						

**3.1.2. a)** Quina és la taxa de fallades ( $T_f$ ) ?

**3.1.2. b)** Suposem que el temps d'accés a la memòria cau, o temps d'accés en cas d'encert ( $t_e$ ), és de 5 ns i el temps total d'accés en cas de fallada ( $t_f$ ) és de 30 ns. Considerant la taxa de fallades obtinguda a la pregunta anterior, quin és el temps mitja d'accés a memòria ( $t_m$ ) ?

## Examen 2

Assignatura	Codi	Data	Hora inici
Estructura de computadors	05.573		

### 3.2 Sistema d'E/S

#### 3.2.1 E/S per interrupcions

Es vol analitzar el rendiment de la comunicació de dades entre la memòria d'un processador i un port USB, utilitzant E/S per interrupcions, amb les següents característiques:

- Velocitat de transferència del dispositiu d'E/S  $v_{\text{transf}} = 1 \text{ MBytes/s} = 1000 \text{ Kbytes/s}$
- Temps de latència mitjà del dispositiu  $t_{\text{latència}} = 0$
- Adreces dels **registres d'estat** i **dades** del controlador d'E/S: 0B00h i 0B04h
- El bit del **registre d'estat** que indica que el controlador del port d'E/S està disponible és el bit 2, o el tercer bit menys significatiu (quan val 1 indica que està disponible)
- Processador amb una freqüència de rellotge de 1 GHz, el temps de cicle  $t_{\text{cicle}} = 1 \text{ ns}$ . El processador pot executar 4 instruccions per cicle de rellotge
- Transferència de **lectura** des del port d'E/S cap a memòria
- Transferència de  $N_{\text{dades}} = 100000$  dades
- La mida d'una dada és  $m_{\text{dada}} = 4 \text{ bytes}$
- El temps per atendre la interrupció ( $t_{\text{rec\_int}}$ ) és de 2 cicles de rellotge

a) Completeu el següent codi CISC que és una rutina de servei a les interrupcions (RSI) per a transferir a través del dispositiu d'E/S anterior, mitjançant la tècnica de E/S per interrupcions.

Es fa servir una variable global que es representa amb l'etiqueta **Addr**, i que al principi del programa conté l'adreça inicial de memòria on emmagatzemar les dades rebudes.

```

1.  _____
2.  PUSH R0
3.  PUSH R1
4.  _____ R0, [0B04h]
5.  MOV R1, [Addr]
6.  MOV [R1], _____
7.  ADD R1, _____
8.  MOV [Addr], _____
9.  POP R1
10. POP R0
11. STI
12. IRET

```

## Examen 2

Assignatura	Codi	Data	Hora inici
Estructura de computadors	05.573		

b) Quant temps dura la transferència del bloc de dades  $t_{\text{transf\_bloc}}$ ?

c) Quin és el percentatge d'ocupació del processador? Percentatge que representa el temps de transferència del bloc  $t_{\text{transf\_bloc}}$  respecte al temps de transferència del bloc per part del perifèric  $t_{\text{bloc}}$

### 3.2.2 Qüestions sobre E/S

a) Quins són els passos principals que s'han de seguir per a garantir que una transferència d'E/S es faci amb èxit?

## Examen 2

Assignatura	Codi	Data	Hora inici
Estructura de computadors	05.573		

**b)** En un sistema d'E/S gestionat per interrupcions. Quina funció tenen les senyals INT i INTA?  
Qui les activa i perquè?

**c)** En un sistema d'E/S gestionat per DMA. Explica quan i perquè es produeix una interrupció.  
Serveix per indicar l'inici o el final d'una transferència? Qui la genera?