

Examen 2013/14-1

Assignatura	Codi	Data	Hora inici
Estructura de computadors	05.573	18/01/2014	12:00

05.573R18R01R14REE9E
05.573 18 01 14 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 2013/14-1

Assignatura	Codi	Data	Hora inici
Estructura de computadores	05.573	18/01/2014	12:00

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: 20%

3.1.1: 10%

3.1.2: 10%

3.2: 20%

3.2.1: 10%

3.2.2: 10%

Examen 2013/14-1

Assignatura	Codi	Data	Hora inici
Estructura de computadors	05.573	18/01/2014	12:00

Pregunta 1

1.1

```

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
; Posicionar el cursor a la pantalla, dins el tauler, en funció de
; la fila(r) i la columna(c) rebudes com a paràmetre sobre els
; registres dil i sil, respectivament, a partir dels valors de
; les variables rowScreenIni i colScreenIni.
; Per calcular la posició del cursor a pantalla (rowCur, colCur)
; utilitzar aquestes fórmules:
; rowCur = rowScreenIni + 3 + (r * 2)
; colCur = colScreenIni + 4 + (c * 4)
; Per posicionar el cursor a la posició (rowCur, colCur) de la pantalla
; s'ha d'utilitzar la subrutina gotoxy.
;
; Variables utilitzades:
; rowScreenIni i colScreenIni
;
; Paràmetres d'entrada :
; rdi: (dil) fila
; rsi: (sil) columna
;
; Paràmetres de sortida:
; Cap
;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
posCurScreen:
    push rbp
    mov  rbp, rsp
    ; guardem els registres que modifiquem i que no són paràm. de sortida.
    push rsi
    push rdi

; escriure el codi per calcular el valor que hem
d'assignar a edi(fila)=rowScreenIni+3+(r*2) i a
esi(columna)=colScreenIni+4+(c*4) per a posicionar el
cursor cridant la subrutina gotoxy.
    shl  dil, 1                ; (r*2)
    add  dil, 3                ; (r*2) + 3
    add  dil, [rowScreenIni] ; (r*2) + 2 + rowScreenIni
    shl  sil, 2                ; (c*4)
    add  sil, 4                ; (c*4) + 4
    add  sil, [colScreenIni] ; (c*4) + 2 + colScreenIni

; Paràmetres d'entrada : (edi) fila, (esi) columna
; Paràmetres de sortida: Cap.
    call gotoxy

posCurScreen_end:
    pop  rdi
    pop  rsi
    mov  rsp, rbp
    pop  rbp
    ret

```

Examen 2013/14-1

Assignatura	Codi	Data	Hora inici
Estructura de computadors	05.573	18/01/2014	12:00

1.2

```

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
; Mostrar els valors de la matriu pieces a la pantalla, dins el tauler,
;   a les posicions corresponents.
; Actualitzar els moviments a la pantalla, en funció la variable moves.
;   NOTA: La variable moves indica els moviments restants, per mostrar
;   el caràcter associat a un valor entre 0 i 9, s'ha de sumar 48
;   (codi ASCII de 0) a aquest valor.
; Posicionar el cursor en el tauler en funció de les variables row i col.
;
; Per a posicionar el cursor utilitzar la subrutina posCurScreen, per a
; accedir a la matriu pieces utilitzar la subrutina getMatrix i per a
; escriure els caràcters a pantalla utilitzar la subrutina printch.
;
; Variables utilitzades: pieces, moves, row i col
; Paràmetres d'entrada: Cap
; Paràmetres de sortida: Cap
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
updateBoard:
...
mov rax, 0
mov rbx, 0
mov rdx, 0
mov rdi, 0
mov rsi, 0

mov bh, 0 ;files
updateBoard_for1:
    cmp bh, dimMatrix
    jge updateBoard_moves

    mov bl, 0 ;columnes
    updateBoard_for2:
        cmp bl, ____dimMatrix____
        jge updateBoard_endfor2

        mov al, bh
        mov ____dil____, al
        mov ____sil____, bl

        ;Paràmetres d'entrada: (dil) fila, (sil) columna
        ;Paràmetres de sortida: Cap
        call posCurScreen

        mov edx, pieces
        call ____getMatrix____ ;al=pieces[bh][bl]
        call printch
        inc ____bl____
        jmp updateBoard_for2

    updateBoard_endfor2:
        inc ____bh____
        jmp updateBoard_for1

updateBoard_moves:
...
updateBoard_end:
...
ret

```

Examen 2013/14-1

Assignatura	Codi	Data	Hora inici
Estructura de computadors	05.573	18/01/2014	12:00

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 = 0000000Ah R1 = 00000001h R2 = 0000F000h	M(00000010h) = 0000F000h M(00000020h) = 0000F000h M(000000A0h) = 0000000Ah	Z = 1, C = 1, S = 0, V = 0
--	--	----------------------------

L'adreça simbòlica W val 00000010h. Quin serà l'estat del computador després d'executar cada fragment de codi? (només modificacions, excloent-hi el PC).

a)	b)
SUB R1, [W+0010h] MOV [000000A0h], R2	ADD R1, [00000020h] XOR [000000A0h], R0
R1 = FFFF1001h M(000000A0h) = 0000F000h Z = 0, S = 1, C = 1, V = 0	R1 = 0000F001h M(000000A0h) = 00000000h Z = 1, S = 0, C = 0, V = 0

2.2

Donat el següent codi en alt nivell:

```
for (i=0,j=7; j>3; i++, j--) {
    V[i]= V[j];
}
```

V1 és un vector de 8 elements de 4 bytes cadascun. Es proposa la següent traducció a CISCA on hem deixat 6 llocs per omplir:

```
INI:  MOV R1, 28           ; Has d'especificar el segon operand
      MOV R3, V
REP:  MOV R4, [V+R1]        ; Has d'especificar el segon operand
      MOV [R3], R4          ; Has d'especificar el primer operand
      ADD R3, 4
      SUB R1, 4              ; Has d'especificar l'instrucció
      CMP R1, 12           ; Has d'especificar el segon operand
      JG REP                 ; Has d'especificar l'instrucció.
```

Examen 2013/14-1

Assignatura	Codi	Data	Hora inici
Estructura de computadors	05.573	18/01/2014	12:00

2.3

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

```
MOV R5, V
ADD R5, 4
NOT [R5]
```

Traduïu-ho a llenguatge màquina i expresseu-ho en la següent taula. Supposeu que la primera instrucció del codi es troba a partir de l'adreça **0CC75BB0h** (que és el valor del PC abans de començar l'execució del fragment de codi). Supposeu també que l'adreça simbòlica V val **0020AA10h**. En la taula de resultats useu una fila per 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-ho 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 a ajuda les taules de codis:

Taula de codis d'instrucció

B0	Instrucció
10h	MOV
20h	ADD
34h	NOT

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.

Examen 2013/14-1

Assignatura	Codi	Data	Hora inici
Estructura de computadors	05.573	18/01/2014	12:00

		Bk per a k=0..10											
@	Assemblador	0	1	2	3	4	5	6	7	8	9	10	
0CC75BB0h	MOV R5, V	10	15	00	10	AA	20	00					
0CC75BB7h	ADD R5, 4	20	15	00	04	00	00	00					
0CC75BBEh	NOT [R5]	34	35										
0CC75BC0h													

Examen 2013/14-1

Assignatura	Codi	Data	Hora inici
Estructura de computadors	05.573	18/01/2014	12:00

Pregunta 3

3.1

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 a una adreça múltiple de 8. Així, el bloc 0 conté les adreces 0, 1, 2, 3, 4, 5, 6 i 7; el bloc 1, les adreces 8, 9, 10, 11, 12, 13, 14 i 15, i el bloc N les adreces $8*N$, $8*N+1$, $8*N+2$, $8*N+3$, $8*N+4$, $8*N+5$, $8*N+6$ i $8*N+7$.

Suposem que el sistema també disposa d'una memòria cau de 4 línies (on cada línia té la mida d'un bloc, és 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 i 7).

3.1.1 Memòria Cau d'Accés Directe

Suposem que el sistema fa servir una **política d'assignació directa**, de manera que cada bloc de la memòria principal només es pot portar a una línia determinada de la memòria cau.

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

30, 22, 28, 5, 6, 7, 8, 9, 17, 40, 55, 12, 22, 53, 39, 13, 21, 32, 22, 41

3.1.1.a) La següent taula mostra l'estat inicial de la cau, que conté les primeres 32 paraules de la memòria (organitzades en 4 blocs). Completar la taula per a mostrar l'evolució de la cau durant l'execució del programa. Per a cada fallada en la cau cal omplir una nova columna indicant quina referència a memòria ha provocat la fallada i el canvi que es produeix en l'estat de la memòria cau (la línia que es modifica).

	Estat Inicial	Fallada: 40
Línia 0	0, 1, 2, 3, 4, 5, 6, 7	0, 1, 2, 3, 4, 5, 6, 7
Línia 1	8, 9, 10, 11, 12, 13, 14, 15	40, 41, 42, 43, 44, 45, 46, 47
Línia 2	16, 17, 18, 19, 20, 21, 22, 23	16, 17, 18, 19, 20, 21, 22, 23
Línia 3	24, 25, 26, 27, 28, 29, 30, 31	24, 25, 26, 27, 28, 29, 30, 31

	Fallada: 55	Fallada: 12
Línia 0	0, 1, 2, 3, 4, 5, 6, 7	0, 1, 2, 3, 4, 5, 6, 7
Línia 1	40, 41, 42, 43, 44, 45, 46, 47	8, 9, 10, 11, 12, 13, 14, 15
Línia 2	48, 49, 50, 51, 52, 53, 54, 55	48, 49, 50, 51, 52, 53, 54, 55
Línia 3	24, 25, 26, 27, 28, 29, 30, 31	24, 25, 26, 27, 28, 29, 30, 31

Examen 2013/14-1

Assignatura	Codi	Data	Hora inici
Estructura de computadors	05.573	18/01/2014	12:00

	Fallada: 22	Fallada: 53
Línia 0	0, 1, 2, 3, 4, 5, 6, 7	0, 1, 2, 3, 4, 5, 6, 7
Línia 1	8, 9, 10, 11, 12, 13, 14, 15	8, 9, 10, 11, 12, 13, 14, 15
Línia 2	16, 17, 18, 19, 20, 21, 22, 23	48, 49, 50, 51, 52, 53, 54, 55
Línia 3	24, 25, 26, 27, 28, 29, 30, 31	24, 25, 26, 27, 28, 29, 30, 31

	Fallada: 39	Fallada: 21
Línia 0	32, 33, 34, 35, 36, 37, 38, 39	32, 33, 34, 35, 36, 37, 38, 39
Línia 1	8, 9, 10, 11, 12, 13, 14, 15	8, 9, 10, 11, 12, 13, 14, 15
Línia 2	48, 49, 50, 51, 52, 53, 54, 55	16, 17, 18, 19, 20, 21, 22, 23
Línia 3	24, 25, 26, 27, 28, 29, 30, 31	24, 25, 26, 27, 28, 29, 30, 31

	Fallada: 41	Fallada:
Línia 0	32, 33, 34, 35, 36, 37, 38, 39	
Línia 1	40, 41, 42, 43, 44, 45, 46, 47	
Línia 2	16, 17, 18, 19, 20, 21, 22, 23	
Línia 3	24, 25, 26, 27, 28, 29, 30, 31	

3.1.1.b) Quina és la taxa de fallades (T_f) ?

$$T_f = 8 \text{ fallades} / 20 \text{ accessos} = 0,40$$

3.1.1.c) Suposem que el temps d'accés a la memòria cau, o temps d'accés en cas d'encert (t_e), és de 2 ns i el temps total d'accés en cas de fallada (t_f) és de 20 ns. Considerant la taxa de fallades obtinguda a la pregunta anterior, quin és el temps mitja d'accés a memòria (t_m) ?

$$t_m = T_f \times t_f + (1 - T_f) \times t_e = 0,4 * 20 \text{ ns} + 0,6 * 2 \text{ ns} = 8 \text{ ns} + 1,2 \text{ ns} = 9,2 \text{ ns}$$

3.1.2 Memòria Cau d'Accés Completament Associatiu

Ara suposem que el mateix 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 LRU, de manera que traurem de la memòria cau aquell bloc que fa més temps que no es referència.

Considerem la mateixa llista de lectures a memòria:

30, 22, 28, 5, 6, 7, 8, 9, 17, 40, 55, 12, 22, 53, 39, 13, 21, 32, 22, 41

Examen 2013/14-1

Assignatura	Codi	Data	Hora inici
Estructura de computadors	05.573	18/01/2014	12:00

3.1.2.a) La següent taula mostra l'estat inicial de la cau, que conté les primeres 32 paraules de la memòria (organitzades en 4 blocs). Completar la taula.

	Estat Inicial	Fallada: 40
Línia 0	0, 1, 2, 3, 4, 5, 6, 7	0, 1, 2, 3, 4, 5, 6, 7
Línia 1	8, 9, 10, 11, 12, 13, 14, 15	8, 9, 10, 11, 12, 13, 14, 15
Línia 2	16, 17, 18, 19, 20, 21, 22, 23	16, 17, 18, 19, 20, 21, 22, 23
Línia 3	24, 25, 26, 27, 28, 29, 30, 31	40, 41, 42, 43, 44, 45, 46, 47

	Fallada: 55	Fallada: 39
Línia 0	48, 49, 50, 51, 52, 53, 54, 55	48, 49, 50, 51, 52, 53, 54, 55
Línia 1	8, 9, 10, 11, 12, 13, 14, 15	8, 9, 10, 11, 12, 13, 14, 15
Línia 2	16, 17, 18, 19, 20, 21, 22, 23	16, 17, 18, 19, 20, 21, 22, 23
Línia 3	40, 41, 42, 43, 44, 45, 46, 47	32, 33, 34, 35, 36, 37, 38, 39

	Fallada: 41	Fallada:
Línia 0	40, 41, 42, 43, 44, 45, 46, 47	
Línia 1	48, 49, 50, 51, 52, 53, 54, 55	
Línia 2	16, 17, 18, 19, 20, 21, 22, 23	
Línia 3	32, 33, 34, 35, 36, 37, 38, 39	

3.1.2.b) Quina és la taxa de fallades (T_f) ?

$$T_f = 4 \text{ fallades} / 20 \text{ accessos} = 0,2$$

3.1.2.c) Suposem que el temps d'accés a la memòria cau, o temps d'accés en cas d'encert (t_e), és de 2 ns i el temps total d'accés en cas de fallada (t_f) és de 20 ns. Considerant la taxa de fallades obtinguda a la pregunta anterior, quin és el temps mitja d'accés a memòria (t_m) ?

$$t_m = T_f \times t_f + (1 - T_f) \times t_e = 0,2 * 20 \text{ ns} + 0,8 * 2 \text{ ns} = 4 \text{ ns} + 1,6 \text{ ns} = 5,6 \text{ ns}$$

3.2

Es vol realitzar la següent comunicació de dades entre la memòria d'un computador i un port USB, que tenen les següents característiques:

Examen 2013/14-1

Assignatura	Codi	Data	Hora inici
Estructura de computadors	05.573	18/01/2014	12:00

- La CPU funciona amb un rellotge de 2GHz de freqüència i executa 1 instrucció per cada 1 cicles de rellotge
- Adreces dels **registres de dades i d'estat** del controlador d'E/S: 0300h i 0304h
- El bit del **registre d'estat** que indica que el controlador del port d'E/S està disponible és el bit 3, o el quart bit menys significatiu (quan val 1 indica que està disponible)
- Transferència d'**escriptura** des de memòria al port d'E/S
- Transferència de $N_{dades}=400.000$ dades, és a dir, $400.000 \times 4 \text{ Bytes} = 1.600.000 \text{ Bytes}$
- Adreça inicial de memòria on resideixen les dades: 0A000000h
- La velocitat de transferència del port és de 20.000 Bytes per segon

3.2.1 E/S programada

Completar el següent codi realitzat amb el repertori CISCA que realitza la transferència descrita abans mitjançant la tècnica d'E/S programada.

```

1.      MOV    R3, 400000
2.      MOV    R2, 0A000000h
3. Bucle: IN     R0, [_0304h_]      ; llegir 4 bytes
4.      AND    R0, _00001000b_
5.      _JE_   Bucle
6.      MOV    R0,[_R2_]           ; llegir 4 bytes
7.      ADD    _R2_, 4
8.      OUT    0300h, _R0_         ; escriure 4 bytes
9.      SUB    R3, __1__
10.     _JNE_  Bucle

```

Quin és el percentatge de temps que dedica la CPU a la tasca d'Entrada/Sortida?

100%

3.2.2 E/S per Interrupcions

Completar el següent codi CISCA que és una rutina de servei a les interrupcions (RSI) per a transferir a través del dispositiu d'E/S anterior, el mateix nombre de dades que abans amb E/S programada, però ara mitjançant la tècnica de E/S per interrupcions. Suposeu:

- Es fa servir una variable global que es representa amb l'etiqueta **Dir**, i que al principi del programa conté l'adreça inicial de memòria on resideixen les dades a transferir

```

1.      _CLI_
2.      PUSH   _R0_
3.      _PUSH_  R1
4.      MOV    _R1_, [Dir]
5.      MOV    _R0_, [R1]
6.      OUT    [_0300h], R0 ; escriure 4 bytes
7.      ADD    R1, _4_
8.      MOV    [_Dir], R1
9.      POP    _R1_
10.     _POP_   R0
11.     STI
12.     IRET

```

Examen 2013/14-1

Assignatura	Codi	Data	Hora inici
Estructura de computadors	05.573	18/01/2014	12:00

Quin és el percentatge de temps que dedica la CPU a la tasca d'Entrada/Sortida?

1.600.000 Bytes a transferir. 20.000 Bytes per segon. Això fa que el temps total de la transferència sigui de 80 segons.

Cada cicle de rellotge és de 0,5ns. Per tant, cada instrucció triga 0,5 ns.

Una interrupció necessita 12 instruccions, per tant són 6 ns.

Hi ha 400.000 interrupcions, per tant són 2.400.000 ns. o 2,4 ms.

Això representa un 0,003% del temps total de la transferència.

Examen 2013/14-1

Assignatura	Codi	Data	Hora inici
Estructura de computadors	05.573	18/01/2014	12:00