

Examen de laboratori d'EC

16 de desembre de 2020 – De 18:05h a 20:00h

INSTRUCCIONS IMPORTANTES:

Baixa els fitxers plantilla de la web d'exàmens

1. La imatge Linux que cal usar hauria d'estar precarregada ja quan comences a fer l'examen. Si no ho estigués, cal usar la imatge "**Linux**" que apareix al menú de possibles imatges.
2. Entra al teu compte personal amb el teu nom d'usuari d'estudiant.
3. Obre el navegador a la pàgina: `https://examens.fib.upc.edu`
4. Clica *Accedir a l'aplicació d'exàmens*, i autentifica't de nou amb el teu nom d'usuari d'estudiant.
5. Busca la pràctica *Examen de Laboratori d'EC - Quadrimestre tardor 2020/21 – Subgrup 41*
6. Obre el fitxer adjunt **examlab-202021q1-subgrup41.zip**, i extrau els fitxers a la teva carpeta. Comença l'examen.

Resol l'examen

1. Posa nom i cognoms a la capçalera dels fitxers següents i que s'han de lliurar de forma individual:
matrius.s, crides.s, respostes_memoria_cache.txt
2. No canviïs el nom dels fitxers.
3. Pots pujar cada fitxer a la web múltiples cops, no esperis a l'últim minut.
4. L'avaluació de cada apartat tindrà en compte principalment el seu funcionament correcte. Es practica-
ran diverses comprovacions, a més a més de la que suggereix el propi enunciat.
5. Per executar el simulador escriu la següent comanda en una finestra de terminal:

```
java -jar /assig/ec/Mars/Mars.jar &
```

Puja les solucions a la web d'exàmens

1. Busca al final de la pàgina web el quadre *Lliurar una nova pràctica*, i verifica que la llista desplegable té seleccionada la pràctica: *Examen de laboratori d'EC - Quadrimestre tardor 2020/21 – Subgrup 41*
2. Clica el botó *Browse* i puja d'un en un els fitxers a lliurar amb la solució:
matrius.s, crides.s, respostes_memoria_cache.txt
3. Verifica que cada fitxer que has pujat no és un fitxer plantilla buit (comprovant la mida en bytes!) i que els noms són els corresponents a aquest torn d'examen.
4. Surt de la sessió que tinguis oberta però **NO APAGUIS EL PC**.

Problema 1. (3,5 punts)

Donat el següent codi en C, escriu al fitxer **matrius.s** el programa equivalent en assemblador de MIPS.

```
int MAT[4][4] = { 9, -1, -3, -2,
                  1, -6, 9, 12,
                  15, 2, 8, 2,
                  19, 6, -2, 3};

main()
{
    int i, j, tmp;

    for (i=0; i<4; i++)
        for (j=3; j>i-1; j--)
        {
            if ((MAT[i][j]%3) != 0)
                tmp = MAT[i][j] & MAT[j][i];
            else
                tmp = MAT[i][j] * 17;

            MAT[j][i] = tmp;
        }
}
```

Tradueix al fitxer **matrius.s** el codi de `main`. Les variables globals ja estan declarades i inicialitzades. Feu servir el registre `$t0` per emmagatzemar la variable `i`, i `$t1` per emmagatzemar la variable `j`.

Comprova que al final de l'execució del programa la matriu global `MAT` conté els valors següents:

```
int MAT[4][4] = { 153, -1, -3, -2,
                  1, -102, 9, 12,
                  -51, 153, 8, 2,
                  18, 204, 2, 51};
```

Problema 2. (3,5 punts)

Tradueix a assembleador MIPS en el fitxer **crides.s** el codi de la funció *subr*. Es valorarà fer servir el menor nombre de registres segurs i seguir les regles de l'ABI.

```
short subr(int p0, unsigned char p1, short *p2) {
    short v1; short v2[3]; short v3; short v4;

    if (p1 >= 127) {
        v1 = 1; v4 = 2;
    } else {
        v1 = 3; v4 = 4;
    }

    v2[0] = *p2;
    v2[1] = *(p2+p0);
    v2[2] = v1;

    v3 = suma(v2);

    *(p2+3) = v3;

    return v3 + v4 + *p2;
}
```

La funció `short suma(short v[3]);` retorna la suma dels 3 elements del vector passat com a paràmetre. El codi de la subrutina `suma` ja està programat i no es pot modificar.

Sabent que el programa principal té declarat:

```
short res1, res2;
short vs[10] = { 8, -6, 4, -2, 2, -4, 6, -8, 10, -10};
```

i que fa 2 crides a `subr`:

```
res1 = subr(1, 0, &vs[0]);
res2 = subr(2, 255, &vs[2]);
```

Verifica que els resultats que s'obtenen en aquestes dues crides són *res1=17 (0x0011)* i *res2=13 (0x000D)*. I que el vector global `vs` al final de l'execució conté els següents valors:

```
short vs[10] = { 8, -6, 4, 5, 2, 7, 6, -8, 10, -10};
```

Problema 3. (3 punts)

Considera un sistema computador format per un processador MIPS, una memòria principal (MP) i una memòria cache de dades (MC), amb la següent configuració:

- correspondència directa
- mida del bloc: 16 bytes (4 words)
- capacitat: 128 bytes (8 blocs)
- escriptura immediata amb assignació

Considera el següent programa en C (que s'ha traduït a MIPS en el fitxer **codi_mem_cache.s**):

```
int V[4];
int M[12][4];

main() {
    int i, j, aux; /* es guarden als registres $t0, $t1, $t2 */

    for (i=0; i<12; i++)
        for (j=0; j<4; j++) {
            aux = M[i][j];
            V[j] = V[j] + aux;
        }
}
```

Suposa que les variables globals V i M s'emmagatzemen en posicions consecutives a partir de l'adreça 0×10010000 i que la MC està inicialment buida. Respon (al fitxer **respostes_memoria_cache.txt**):

a) (0,5 punts)

Quants blocs ocupa V ?

Quants blocs ocupa M ?

Quin número de bloc de la cache pot contenir tant elements de V com de M ?

Indica quins elements de V i M es mapegen en el mateix número de bloc de la cache:

Elements de V (llista d'elements):

Elements de M (llista d'elements):

b) (1 punt)

Quin és el percentatge d'encert d'aquests programa en aquesta cache:

Quantes fallades de cache són causades per accessos a V per lectura:

Quantes fallades de cache són causades per accessos a M per lectura:

Quantes fallades de cache són causades per accessos a V per escriptura:

c) (0,5 punts)

Quins tipus de localitat tenen els accessos a V (*Espacial, temporal, Espacial i temporal*):

Quins tipus de localitat tenen els accessos a M (*Espacial, temporal, Espacial i temporal*):

d) (1 punt)

Suposa que volem millorar la taxa d'encerts sense variar la capacitat total de la cache (128 bytes), solament podem modificar el grau d'associativitat (A) i la llargada del bloc (B).

El mínim nombre de fallades s'obté amb la combinació:

Grau d'associativitat (núm de vies) A:

Mida del bloc (núm de words) B:

Nombre total de fallades:

Quin és el percentatge d'encert d'aquets programa en aquesta nova organització de la cache: