

Segundo-parcial.pdf



AlbertoM02



Fundamentos de Computadores y Redes



1º Grado en Ingeniería Informática del Software



Escuela de Ingeniería Informática Universidad de Oviedo







El coche lo ponemos nosotros, tú solo tienes que escoger al copiloto que haga de DJ y de GPS a la vez.





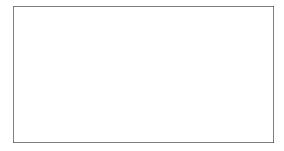
Examen de Fundamentos de Computadores y Redes

Teoría. Segundo examen parcial. Curso 2020-2021

Cada respuesta incorrecta, ilegible o vacía no suma ni resta.

1 ☐ (1 punto) El siguiente fragmento de código C++ debe traducirse al ensamblador del CT. Indica las instrucciones necesarias teniendo en cuenta que la variable p se almacena en el registro r0, el vector vt a partir de la dirección de memoria 9825h y que el registro r1 ya contiene el valor 3.

```
int p = -9;
vt[3]++;
```



2 ☐ (1 punto) El siguiente fragmento de código C++ debe traducirse al ensamblador del CT. Indica las instrucciones necesarias teniendo en cuenta que las variables f, a y d son naturales y se almacenan en los registros r1, r4 y r2, respectivamente.

```
if (f >= (a + d))
  f = f + a; // f = f + a
```



3 ☐ (1 punto) Se ha escrito el siguiente fragmento de código en C++. Sabiendo que la variable j se almacena en r4 y que la variable p se almacena en r3, traduce el código al lenguaje del

```
unsigned int j = 120;
for (unsigned int p = 140; p == j; p--)
  j = j++;
```

4 ☐ (1 punto) Se ha escrito el siguiente fragmento de código en

```
c1 = MyResult(a1, b1);
```

Tradúcelo al lenguaje del CT teniendo en cuenta que la variable c1 se almacena en el registro r2; la variable a1 se almacena en el registro r3; la variable b1 se almacena en memoria en la posición 192Ah y se pasa por referencia; el paso de parámetros se realiza a través de la pila; el orden de paso de parámetros es de derecha a izquierda y el procedimiento devuelve el resultado en



coches.net





5 ☐ (2 puntos) Se tiene el siguiente procedimiento en C++:

```
unsigned int NewProc(unsigned int p1, unsigned int& p2)
 if (p2 >= p1)
   return 0;
 else
   return 1;
```

Sabiendo que el paso de parámetros se realiza a través de la pila CT.

```
de derecha a izquierda, que uno de los parámetros se pasa por
valor y otro por referencia y que el procedimiento devuelve su
valor en r0, traduce el procedimiento anterior al lenguaje del
```

6 □ Se tiene siguiente fragmento en el lenguaje del CT:

```
mov r0, r4
loop:
  cmp r1, r2
  brc next
  and r3, r5, r0
  neg r0
  inc r2
  jmp loop; --INSTR1--
next:
  neg r6
  push r1; -- INSTR2--
```

El CT acaba de terminar la ejecución de una instrucción (no perteneciente al código anterior) y va a comenzar a ejecutar el código anterior. Se conoce el valor de los registros del CT: R0=000Fh R1=0A42h R2=0A42h R3=010Eh R4=FAA9h R5=0CC8h R6=0CC8h R7=AB84h PC=F193h IR=3600h Recuerda que el número que va a continuación de 1239h es

123Ah, y el número que va antes de 1210h es 120Fh.

a — (0,5 puntos) ¿Cuál es el mnemónico de la instrucción que se ha terminado de ejecutar?

b— (**0,5 puntos**) ¿Cuál sería la codificación de la instrucción marcada como -INSTR1- si se sustituyese por la instrucción JMP there y la etiqueta there marcase la posición de memoria F192h? (Responder en hexadecimal)

c — (0,5 puntos) Cuando se ejecute la instrucción marcada como -INSTR2-, ¿a qué dirección de memoria se accede después del paso 4 de su ejecución y antes de su finalización? (Responder en hexadecimal)

d— (1 punto) Suponiendo que el CT emplea un reloj de 50 MHz ¿cuánto tiempo tarda en ejecutarse el fragmento de programa anterior? (Responder en μ s)

7 🗆	(0,5 puntos) Se ha	añadido al	Computador	Teórico	la siguiente
iı	nstrucción:				

Código instrucción	Mnemónico	Operación
0111000 Ri Rs1 Rs2	ADD [Ri], Rs1, Rs2	$[Ri] \leftarrow Rs1 + Rs2$

¿Cuál sería el código que le corresponde a la instrucción ADD [R7], R4, R5? (Contesta en hexadecimal).

8 ☐ (1 punto) Se desea añadir una nueva instrucción para el C
con el mnemónico MOVEX [R7], Inm8. Esta instrucción copi
a la posición de memoria apuntada por el registro R7 el valo
Inm8 transformado a 16 bits mediante extensión de signo. Indi
car la secuencia de señales de control para los pasos 4 y siguien
tes necesarias para su ejecución en el menor número de paso
posibles.

Paso	Señales

