Certamen 2 Arquitectura de Computadores **(Me saqué un 58 xD)**

Nombre: **Pablo Montoya Gutiérrez**.

Semestre 2019/2

1. Defina con sus palabras los siguientes conceptos (18 pts)

a. Segmento de Instrucciones: **En esta sección podemos encontrar el código del archivo con su main y funciones. La podemos encontrar en el código como “.text”.**

b. Segmento de Datos: **En esta sección nos encontramos con las variables del programa como mensajes, enteros, flotantes, variables globales, etc. La podemos encontrar en el código como “.data”.**

c. Registro de dirección de retorno ($RA): **(Return Address) Nos guarda el valor de retorno de una función el cual es guardado una vez que se termine la ejecución de la función llamada.**

d. Stack o Pila: **Es una estructura de datos el cual sigue el método de ordenamiento LIFO (Last In First Out),gracias a ella podemos conservar datos de las instrucciones que se van ejecutando.**

e. Dirección de memoria: **Se tienen muchas posiciones de memoria las cuales nos ayudarán a poder almacenar datos de las variables como también resultados de las operaciones y lo que se retorne de las funciones.**

f. Registro contador de programa ($PC): **Nos dice dónde se encuentra el procesador al momento de estar ejecutando el segmento de instrucciones.**

2. Considere el código assembler MIPS que es similar (no igual) al 4.s analizado en el laboratorio

y responda las preguntas que se detallan luego del código. (50 Pts)

Línea Dirección (Hex) Programa 1 #-------- segmento de texto -------- 2 .text 3 .globl main 4 5 0x4400 0000 main: la $a0, x 6 0x4400 0004 la $a1, y 7 0x4400 0008 lw $a2, size 8 0x4400 000C 9 0x4400 0010 li $v0, 0 10 0x4400 0014 li $t3,0 11 0x4400 0018 ip: bge $t3, $a2, fin 12 0x4400 001C lw $t0,0($a0) 13 0x4400 0020 lw $t1,0($a1) 14 0x4400 0024 mul $t2,$t0,$t1 15 0x4400 0028 add $v0,$v0,$t2 16 0x4400 002C addi $a0,$a0,4 17 0x4400 0030 addi $a1,$a1,4 18 0x4400 0034 addi $t3,$t3,1 19 0x4400 0038 b ip 20 0x4400 003C fin: move $a0 $v0 21 0x4400 0040 li $v0,1 22 0x4400 0044 syscall 23 0x4400 0048 li $v0, 10 24 0x4400 004C syscall 25 .... 26 #-------- segmento de datos -------- 27 .data 28 0x2200 0000 size: .word 5 29 0x2200 0004 x: .word 1,2,3,4,5 30 0x2200 0018 y: .word 5,4,3,2,1

a. En el código, que significado poseen las directivas .data (línea 27) y .text (línea 2) (4pts)

**.data: Almacenar las variables tales como .word, .asciiz, variables globales, etc.**

**.text: Directiva donde podemos encontrar el código, programa principal, funciones, instrucciones en general.**

b. ¿Cuál es la dirección de memoria de la variable “size” y de los arreglos “x” e “y”? (4pts)

**Size: 0x2200 0000**

**X: 0x2200 0004**

**Y: 0x2200 0018**

c. ¿En qué dirección de memoria se inicia el programa? (4pts)

**0x44000000**

d. ¿Qué valor tiene el registro $PC cuando se ejecuta la línea 10? (4pts)

**0x00400018**

e. ¿Cuál es el valor (dirección de memoria) de las etiquetas “ip” (línea 11), y “fin” (línea 20)? (4pts)

**ip: 0x44000018**

**fin: 0x4400003C**

f. Explique qué hace el programa y qué valor se imprime en pantalla. (4pts)

**El programa multiplica los valores de la posición i de cada vector ( x[i] \* y[i] ) y luego suma las multiplicaciones imprimiendo en pantalla en valor final “35”**

g. ¿Qué dirección de memoria posee el registro $PC luego que la instrucción de la línea 11

es ejecutada y evaluada como falsa? (4pts)

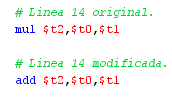
**0x4400001C**

h. ¿Qué dirección de memoria posee el registro $PC luego que la instrucción de la línea 11

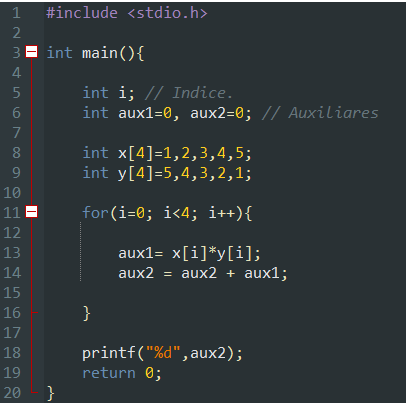
es ejecutada y evaluada como verdadera? (4pts)

**0x4400003C**

i. Modifique el programa para que calcule la suma de las diferencias entre valores de igual índice (es decir: sumar x[i]-y[i]) de los arreglos “x” e “y” (4pts). (Puede hacer la modificación en el mismo programa o bien señale los números de las líneas del código que requieren modificación y cuál sería la modificación)



j. Desarrolle un programa en C equivalente. (14 Pts)



3. Sofanor Rodríguez ltda, empresa innovadora en sistemas computacionales, desea desarrollar un programa en MIPS que calcule las temperaturas máximas y mínimas de un posible caso de corona virus. Para facilitarle su programa, considere que el registro de temperaturas de un paciente ya se almacena en un arreglo de acuerdo con el siguiente esquema: (32 Pts)

.data temperaturas .word 36, 37, 35, 38, 37, 39, 38 len .word 7

La variable “len” señala el largo del arreglo. Los valores máximos y mínimos deben ser almacenados en los registros $v0 y $v1 de MIPS.

