

Arquitectura de Computadoras

Parcial 2

Apellido y Nombre: Nro. de Legajo:

Observaciones: Escribir las respuestas con tinta. Cada ejercicio indica su valor en puntos. **SE APRUEBA CON 10 PUNTOS. NOTA MAXIMA: 20.**

1. Dado el siguiente programa, indicar la cantidad de atascos tipo RAW que se producen, la cantidad de Branch Taken Stalls y el número de CPI, si la ejecución se efectúa sólo con la opción Forwarding habilitada (0,5 pts c/u).

```
.data
tabla1: .word 15, 11, 24
tabla2: .word 0, 0, 0
```

```
.code
DADDI r1, r0, 0
DADDI r2, r0, 3
```

```
LOOP: LD r3, tabla1(r1)
      DADDI r3, r3, 1
      SD r3, tabla2(r1)
      DADDI r1, r1, 8
      DADDI r2, r2, -1
      BNEZ r2, LOOP
      HALT
```

Nro. RAWs: _____ BTS: _____ CPI: ____/____

2. Indique la nueva cantidad de atascos RAW, BTS y Branch Misprediction si el programa anterior se ejecuta ahora con la opción Branch Target Buffer (BTB) habilitada (0,5 pts c/u).

Nro. RAWs: _____ BTS: _____ Branch Misprediction: _____

3. El valor del CPI con BTB habilitado es mayor, igual o menor que en el ejercicio 1? Justifique brevemente (0,5 pts)

4. Reordenar las instrucciones del lazo del programa del ejercicio 1 de manera que, ejecutándolo con la opción Delay Slot habilitada, el resultado generado por el programa sea igual al original (1 pto).

```
LOOP: _____
      _____
      _____
      _____
      _____
      _____
      HALT
```

5. Completar las dos instrucciones necesarias para convertir a entero un número de punto flotante ubicado en el registro f9 y dejarlo en el registro r5 (0,5 pts c/u).

```
_____
_____
```

6. Completar las tres instrucciones para imprimir en la pantalla alfanumérica del simulador un número en punto flotante, ya almacenado en el registro f2 (0,5 pts c/u).

```
.code
lwu $s6, CONTROL($0) _____
lwu $s7, DATA($0) _____
```

7. El siguiente programa recorre un arreglo de números en punto flotante (TABLA1) y genera otro arreglo (TABLA2) con los números que están dentro del rango establecido por MIN y MAX. Además, guarda en RES la cantidad de elementos de TABLA2. Completar el programa con las tres instrucciones faltantes de manera que funcione correctamente (1 pto c/instrucción)

NOTA (del set de instrucciones): c.lt.d F_d, F_r ; compara F_d con F_r , dejando flag FP=1 si F_d es menor que F_r (en punto flotante)
bc1t OFFSET ; salta a la dirección rotulada OFFSET si flag FP=1 (ó true) (en punto flotante)

<pre>.data TABLA1: .double 12.0, 15.5, 31.2, 56.4, 44.3, 78.1 MIN: .double 20.0 MAX: .double 50.0 CANT: .word 6 RES: .word 0 TABLA2: .double 0.0 .text LD \$t0, CANT(\$zero) LD \$t1, MIN(\$zero) _____ DADD \$t3, \$zero, \$zero DADD \$t4, \$zero, \$zero</pre>	<pre>LOOP: DADD \$t5, \$zero, \$zero _____ C.LT.D \$f3, \$f1 BCIT FUERA C.LT.D \$f2, \$f3 BCIT FUERA DADDI \$t3, \$t3, 1 S.D \$f3, TABLA2(\$t5) _____ FUERA: DADDI \$t4, \$t4, 8 DADDI \$t0, \$t0, -1 BNEZ \$t0, LOOP SD \$t3, RES(\$zero) HALT</pre>
--	---

