



Redictado Arquitectura de Computadoras.

2do Parcial. WINMIPS. 1ra Fecha. 19-Jun-2018. TEMA 2

Alumno: [redacted]

Se aprueba con 6 (seis) puntos. Puntaje total: 10 PUNTOS.

1. Cual es el número y tamaño de los registros del WinMips? (1 punto si correcta, -0,5 si incorrecta, 0 si NS/NC)
☒ 64 registros de P. Fijo, y 64 P. Flotante de 32bits c/u ☐ 64 registros de P. Fijo, de 64bits
☐ 32 registros de P. Fijo, y 32 P. Flotante de 64bits c/u ☐ Ns/Nc
2. ¿Qué ventaja presenta la habilitación de la opción Forwarding en la ejecución de un programa dentro de Winmips? (1 punto si correcta, -0,5 puntos si incorrecta, 0 si NS/NC)
☐ Evita atascos estructurales ☐ Reduce el consumo de memoria
☐ Reduce el número de CPIs ☐ Ns/Nc
3. ¿Cuales son las instrucciones de llamada y retorno hacia y desde una subrutina? (1 punto si correcta, -0,5 puntos si incorrecta, 0 si NS/NC)
☐ JAL y R31 ☐ CALL y RET
☐ JAL y JR ☐ Ns/Nc
4. Indique con un tilde o una cruz cuáles de estas instrucciones son correctas o incorrectas? (1 punto)
☒ R7, ETIQUETA ☐ lbu \$s0, VARIABLE(\$0)
☐ sd \$t0, 4(\$s0) ☐ daddi R3, R5, R7
5. Dado el siguiente programa, y con la opción Branch Target Buffer habilitada, indicar cuántos Branch Misprediction Stalls se presentan. (1 punto)

```
loop: daddi r1, r0, 100
      nop
      daddi r1, r1, -1
      bnez r1, loop
      halt
```

6. Si en el ejercicio anterior se ejecuta el mismo programa, pero con la opción Branch Target Buffer deshabilitada, cuántos Branch Misprediction Stalls se presentan ahora? (1 punto).

7. Escriba las instrucciones en WinMips para simular un push a la pila (apuntado por \$sp) del registro \$t2, y luego un pop al registro \$t1 (0.5 puntos c/u)

PUSH:

daddi

sd

POP:

ld

daddi

8. El siguiente programa lee un numero de teclado e informa como resultado el doble de dicho valor. Complete los espacios en blanco. (0,5 puntos c/u)

CONTROL:

DATA:

```
.word32 0x10000
.word32 0x10008
.code
lwu $s0, [redacted]
lwu $s1, [redacted]
daddi $t0, $0, 8
sd $t0, [redacted]
ld $v1, 0($s0)
dadd $v1, $v1, $v1
sd $v1, 0($s0)
daddi $t0, $0, 1
sd $t0, [redacted]
```

```
; configuro $s0 con la dirección de DATA
; configura $s1 con la dirección de CONTROL
; envío a CONTROL la señal para leer un numero de teclado
; duplico el valor ingresado
; envío a CONTROL la señal para imprimir un numero en pantalla
```