นางสาวจิรัชฌา วิทยาชีวะ 6010502519
นางสาวจารวี ทองเจริญ 6010504660

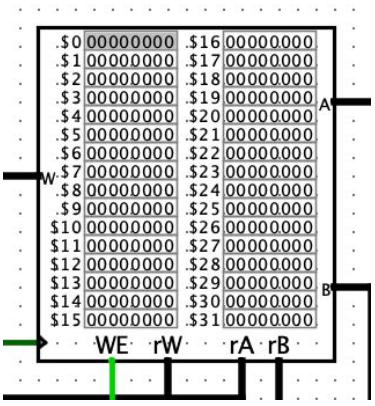
ทดสอบการทำงานของ CPU บน Logicsim ว่าทำงานได้ถูกต้องหรือไม่ (จาก File testcase.s)

Address	Binary	Assembly
		main:
00000000	24080005	addiu $t0, $zero, 5 # $t0 = 0 + 5
00000004	24090007	addiu $t1, $zero, 7 # $t1 = 0 + 7
00000008	24100009	addiu $s0, $zero, 9 # $s0 = 0 + 9
0000000c	24110002	addiu $s1, $zero, 2 # $s1 = 0 + 2
00000010	0c000008	jal func_plus
00000014	00085040	sll $t2, $t0, 1
00000018	00085042	srl $t2, $t0, 1
0000001c	00085043	sra $t2, $t0, 1
		func_plus:
00000020	01095021	addu $t2, $t0, $t1 # $t2 = $t0 + $t1 = 12...
00000024	02119021	addu $s2, $s0, $s1 # $s2 = $s0 + $s1 = 11...
00000028	2d4b0014	sltiu $t3, $t2, 20 # if $t2 < 20 then $t3...
0000002c	11600013	beq $t3, $zero, bistwise_2 # if $t3 != 0 then go ...
00000030	294b0004	slti $t3, $t2, 4 # if $t2 < 4 then $t3 ...
00000034	01095023	subu $t2, $t0, $t1 # $t2 = $t0 - $t1 = -2
00000038	0148582b	sltu $t3, $t2, $t0 # if $t2 < $t0 then $t...
0000003c	1560000a	bne $t3, $zero, bistwise # if $t3 != 0 then go ...
00000040	0109582a	slt $t3, $t0, $t1 # if $t0 < $t1 then $t...
00000044	15600008	bne $t3, $zero, bistwise # if $t3 = 0 then go t...
		load_store:
00000048	27bdfffc	addiu $sp, $sp, -4 # add strack
0000004c	afa80000	sw $t0, 0($sp) # store $t0 in 0($sp)
00000050	27bdfffc	addiu $sp, $sp, -4 # add strack
00000054	afa90000	sw $t1, 0($sp) # store $t1 in 0($sp)
00000058	8fab0004	lw $t3, 4($sp) # load $t0 to $t3
0000005c	8fac0000	lw $t4, 0($sp) # load $t1 to $t4
00000060	27bd0008	addiu $sp, $sp, 8 # return strack
00000064	0800001f	j bistwise_2
		bistwise:
00000068	01096024	and $t4, $t0, $t1
0000006c	01096825	or $t5, $t0, $t1
00000070	01097026	xor $t6, $t0, $t1
00000074	01097827	nor $t7, $t0, $t1
00000078	08000012	j load_store
		bistwise_2:
0000007c	310c000c	andi $t4, $t0, 12
00000080	350d000d	ori $t5, $t0, 13
00000084	390e000e	xori $t6, $t0, 14
00000088	03e00008	jr $ra

ทดสอบการทำงานของ addiu และ addu ใน logic sim (ของ week แรก)

อธิบาย :

ค่าเริ่มต้นทุก registers มีค่าเป็น 0x00000000

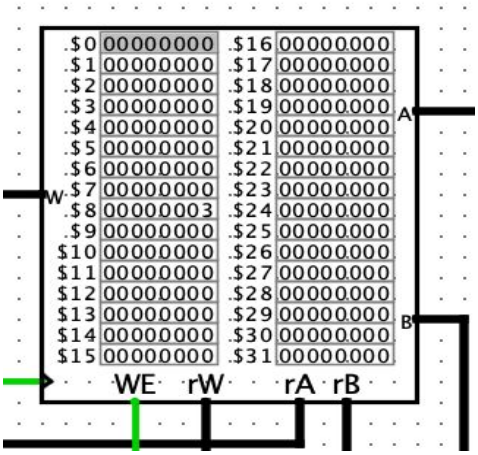


นางสาวจิรัชฌา วิทยาชีวะ 6010502519
นางสาวจารวี ทองเจริญ 6010504660

```
addiu   $t0, $zero, 3
```

อธิบาย :

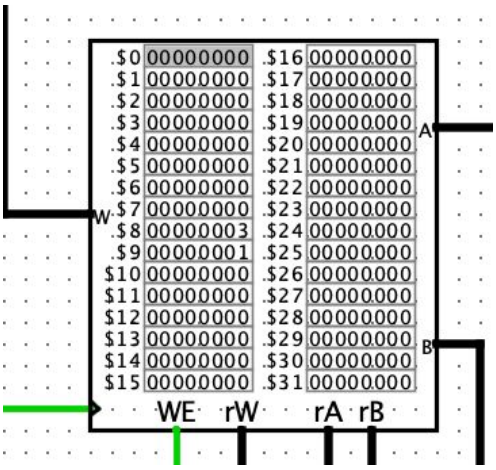
จากคำสั่งข้างต้น จะนำค่า 0 มาบวกกับค่าคงที่ 3 มาใส่ที่ $t0 ดังนั้น register $8 จึงเท่ากับ 0x00000003




```
addiu   $t1, $zero, 1
```

อธิบาย :

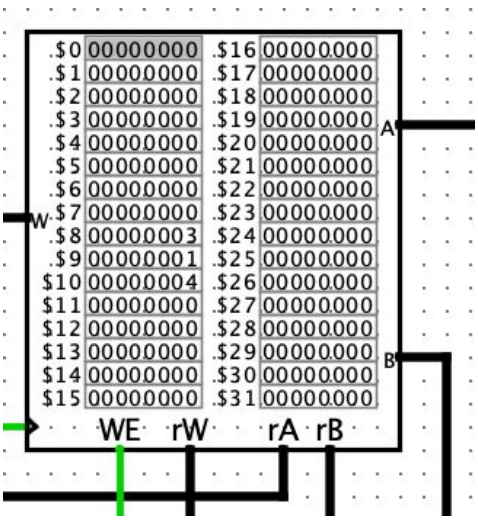
จากคำสั่งข้างต้น จะนำค่า 0 มาบวกกับค่าคงที่ 1 มาใส่ที่ $t1 ดังนั้น register $9 จึงเท่ากับ 0x00000001




```
addu    $t2, $t0, $t1
```

อธิบาย :

จากคำสั่งข้างต้น จะนำค่าจาก register $8 มาบวกกับค่า ค่าจาก register $9 มาใส่ที่ $t2 ดังนั้น register $10 จึงเท่ากับ 0x00000004



นางสาวจิรัชฌา วิทยาชีวะ 6010502519
นางสาวจารวี ทองเจริญ 6010504660

หมายเหตุ: ในส่วนของสัปดาห์แรกยังไม่ได้สร้าง Module ที่รองรับคำสั่งอื่นๆนอกจากคำสั่ง addiu และ addu ดังนั้นจึงขอนำโค้ด testcase.s ไปรันในโปรแกรม QtSpim เพื่อทำการประมวณผลคำสั่งที่ยังไม่ได้สร้าง Module ผลลัพธ์ที่ได้หรือค่าต่างๆใน Register หรือ Memory นั้นขอส่งในรูปของโปรแกรม QtSpim ส่วนในสัปดาห์สอง เมื่อทำ Module เสร็จสิ้นแล้ว จะมาทำการแก้ไข README_test_programs แล้วจึงจะนำส่งอีกรอบค่ะ

Main :

```
addiu   $t0, $zero, 5
```

อธิบาย :

จากคำสั่งข้างต้น จะนำค่า 0 มาบวกกับค่าคงที่ 5 (แบบ unsigned) มาใส่ที่ $t0 ดังนั้น register $8 จึงเท่ากับ 0x00000005

```
R0  [r0] = 0
R1  [at] = 0
R2  [v0] = 4
R3  [v1] = 0
R4  [a0] = 1
R5  [a1] = 7ffffe0c
R6  [a2] = 7ffffe14
R7  [a3] = 0
R8  [t0] = 5
R9  [t1] = 0
R10 [t2] = 0
R11 [t3] = 0
R12 [t4] = 0
R13 [t5] = 0
R14 [t6] = 0
R15 [t7] = 0
R16 [s0] = 0
R17 [s1] = 0
R18 [s2] = 0
R19 [s3] = 0
R20 [s4] = 0
R21 [s5] = 0
R22 [s6] = 0
R23 [s7] = 0
R24 [t8] = 0
R25 [t9] = 0
R26 [k0] = 0
R27 [k1] = 0
R28 [gp] = 10008000
R29 [sp] = 7ffffe08
R30 [s8] = 0
R31 [ra] = 400018
```

```
addiu   $t1, $zero, 7
```

อธิบาย :

จากคำสั่งข้างต้น จะนำค่า 0 มาบวกกับค่าคงที่ 7 (แบบ unsigned) มาใส่ที่ $t1 ดังนั้น register $9 จึงเท่ากับ 0x00000007

```
R0  [r0] = 0
R1  [at] = 0
R2  [v0] = 4
R3  [v1] = 0
R4  [a0] = 1
R5  [a1] = 7ffffe0c
R6  [a2] = 7ffffe14
R7  [a3] = 0
R8  [t0] = 5
R9  [t1] = 7
R10 [t2] = 0
R11 [t3] = 0
R12 [t4] = 0
R13 [t5] = 0
R14 [t6] = 0
R15 [t7] = 0
R16 [s0] = 0
R17 [s1] = 0
R18 [s2] = 0
R19 [s3] = 0
R20 [s4] = 0
R21 [s5] = 0
R22 [s6] = 0
R23 [s7] = 0
R24 [t8] = 0
R25 [t9] = 0
R26 [k0] = 0
R27 [k1] = 0
R28 [gp] = 10008000
R29 [sp] = 7ffffe08
R30 [s8] = 0
R31 [ra] = 400018
```

นางสาวจิรัชฌา วิทยาชีวะ 6010502519
นางสาวจารวี ทองเจริญ 6010504660

```
addiu   $s0, $zero, 9
```

```
R0  [r0] = 0
R1  [at] = 0
R2  [v0] = 4
R3  [v1] = 0
R4  [a0] = 1
R5  [a1] = 7ffffe0c
R6  [a2] = 7ffffe14
R7  [a3] = 0
R8  [t0] = 5
R9  [t1] = 7
R10 [t2] = 0
R11 [t3] = 0
R12 [t4] = 0
R13 [t5] = 0
R14 [t6] = 0
R15 [t7] = 0
R16 [s0] = 9
R17 [s1] = 0
R18 [s2] = 0
R19 [s3] = 0
R20 [s4] = 0
R21 [s5] = 0
R22 [s6] = 0
R23 [s7] = 0
R24 [t8] = 0
R25 [t9] = 0
R26 [k0] = 0
R27 [k1] = 0
R28 [gp] = 10008000
R29 [sp] = 7ffffe08
R30 [s8] = 0
R31 [ra] = 400018
```

อธิบาย :

จากคำสั่งข้างต้น จะนำค่า 0 มาบวกกับค่าคงที่ 9 (แบบ unsigned) มาใส่ที่ $s0 ดังนั้น register $16 จึงเท่ากับ 0x0000009

```
addiu   $s1, $zero, 2
```

```
R0  [r0] = 0
R1  [at] = 0
R2  [v0] = 4
R3  [v1] = 0
R4  [a0] = 1
R5  [a1] = 7ffffe0c
R6  [a2] = 7ffffe14
R7  [a3] = 0
R8  [t0] = 5
R9  [t1] = 7
R10 [t2] = 0
R11 [t3] = 0
R12 [t4] = 0
R13 [t5] = 0
R14 [t6] = 0
R15 [t7] = 0
R16 [s0] = 9
R17 [s1] = 2
R18 [s2] = 0
R19 [s3] = 0
R20 [s4] = 0
R21 [s5] = 0
R22 [s6] = 0
R23 [s7] = 0
R24 [t8] = 0
R25 [t9] = 0
R26 [k0] = 0
R27 [k1] = 0
R28 [gp] = 10008000
R29 [sp] = 7ffffe08
R30 [s8] = 0
R31 [ra] = 400018
```

อธิบาย :

จากคำสั่งข้างต้น จะนำค่า 0 มาบวกกับค่าคงที่ 2 (แบบ unsigned) มาใส่ที่ $s1 ดังนั้น register $17 จึงเท่ากับ 0x0000002

นางสาวจิรัชฌา วิทยาชีวะ 6010502519
นางสาวจารวี ทองเจริญ 6010504660

```
jal func_plus
```

```
R0  [r0] = 0
R1  [at] = 0
R2  [v0] = 4
R3  [v1] = 0
R4  [a0] = 1
R5  [a1] = 7ffffe0c
R6  [a2] = 7ffffe14
R7  [a3] = 0
R8  [t0] = 5
R9  [t1] = 7
R10 [t2] = 0
R11 [t3] = 0
R12 [t4] = 0
R13 [t5] = 0
R14 [t6] = 0
R15 [t7] = 0
R16 [s0] = 9
R17 [s1] = 2
R18 [s2] = 0
R19 [s3] = 0
R20 [s4] = 0
R21 [s5] = 0
R22 [s6] = 0
R23 [s7] = 0
R24 [t8] = 0
R25 [t9] = 0
R26 [k0] = 0
R27 [k1] = 0
R28 [gp] = 10008000
R29 [sp] = 7ffffe08
R30 [s8] = 0
R31 [ra] = 400038
```

อธิบาย :

จากคำสั่งข้างต้น จะโดดไปยังคำสั่ง func_plus และเก็บ address ไว้ที่ $ra ดังนั้น register $31 จึงเท่ากับ 0X400038 ดังรูป

Func_plus:

```
addu $t2, $t0, $t1
```

```
R0  [r0] = 0
R1  [at] = 0
R2  [v0] = 4
R3  [v1] = 0
R4  [a0] = 1
R5  [a1] = 7ffffe0c
R6  [a2] = 7ffffe14
R7  [a3] = 0
R8  [t0] = 5
R9  [t1] = 7
R10 [t2] = c
R11 [t3] = 0
R12 [t4] = 0
R13 [t5] = 0
R14 [t6] = 0
R15 [t7] = 0
R16 [s0] = 9
R17 [s1] = 2
R18 [s2] = 0
R19 [s3] = 0
R20 [s4] = 0
R21 [s5] = 0
R22 [s6] = 0
R23 [s7] = 0
R24 [t8] = 0
R25 [t9] = 0
R26 [k0] = 0
R27 [k1] = 0
R28 [gp] = 10008000
R29 [sp] = 7ffffe08
R30 [s8] = 0
R31 [ra] = 400038
```

อธิบาย :

จากคำสั่งข้างต้น จะนำค่าใน $t0 มาบวกกับค่าใน $t1 (แบบ unsigned) แล้วนำผลลัพธ์มาใส่ที่ register $t2 ดังนั้น register $10 จึงเท่ากับ 0x0000000c

นางสาวจิรัชฌา วิทยาชีวะ 6010502519
นางสาวจารวี ทองเจริญ 6010504660

```
addu $s2, $s0, $s1
```

```
R0  [r0] = 0
R1  [at] = 0
R2  [v0] = 4
R3  [v1] = 0
R4  [a0] = 1
R5  [a1] = 7ffffe0c
R6  [a2] = 7ffffe14
R7  [a3] = 0
R8  [t0] = 5
R9  [t1] = 7
R10 [t2] = c
R11 [t3] = 0
R12 [t4] = 0
R13 [t5] = 0
R14 [t6] = 0
R15 [t7] = 0
R16 [s0] = 9
R17 [s1] = 2
R18 [s2] = b
R19 [s3] = 0
R20 [s4] = 0
R21 [s5] = 0
R22 [s6] = 0
R23 [s7] = 0
R24 [t8] = 0
R25 [t9] = 0
R26 [k0] = 0
R27 [k1] = 0
R28 [gp] = 10008000
R29 [sp] = 7ffffe08
R30 [s8] = 0
R31 [ra] = 400038
```

อธิบาย :

จากคำสั่งข้างต้น จะนำค่าใน $s0 มาบวกกับค่าใน $s1 (แบบ unsigned) แล้วนำผลลัพธ์มาใส่ที่ register $s2 ดังนั้น register $18 จึงเท่ากับ 0x000000b

```
sltiu $t3, $t2, 20
```

```
R0  [r0] = 0
R1  [at] = 0
R2  [v0] = 4
R3  [v1] = 0
R4  [a0] = 1
R5  [a1] = 2147483148
R6  [a2] = 2147483156
R7  [a3] = 0
R8  [t0] = 5
R9  [t1] = 7
R10 [t2] = 12
R11 [t3] = 1
R12 [t4] = 0
R13 [t5] = 0
R14 [t6] = 0
R15 [t7] = 0
R16 [s0] = 9
R17 [s1] = 2
R18 [s2] = 11
R19 [s3] = 0
R20 [s4] = 0
R21 [s5] = 0
R22 [s6] = 0
R23 [s7] = 0
R24 [t8] = 0
R25 [t9] = 0
R26 [k0] = 0
R27 [k1] = 0
R28 [gp] = 268468224
R29 [sp] = 2147483144
R30 [s8] = 0
R31 [ra] = 4194360
```

อธิบาย :

จากคำสั่งข้างต้น จะนำค่าใน $t2 มาเทียบกับค่าคงที่ 20 ว่าค่าใน register $t2 น้อยกว่าหรือไม่ (แบบ unsigned) ถ้าเป็นจริงผลลัพธ์จะมีค่าเท่ากับ 1 แต่ถ้าเป็นเท็จผลลัพธ์จะมีค่าเท่ากับ 0 แล้วนำผลลัพธ์ที่ได้มาใส่ที่ register $t3 ดังนั้น register $11 จึงมีค่าเท่ากับ 0x0000001

นางสาวจิรัชฌา วิทยาชีวะ 6010502519
นางสาวจารวี ทองเจริญ 6010504660

```
beq $t3, $zero, bistwise_2
```

```
R0  [r0] = 0
R1  [at] = 0
R2  [v0] = 4
R3  [v1] = 0
R4  [a0] = 1
R5  [a1] = 2147483148
R6  [a2] = 2147483156
R7  [a3] = 0
R8  [t0] = 5
R9  [t1] = 7
R10 [t2] = 12
R11 [t3] = 1
R12 [t4] = 0
R13 [t5] = 0
R14 [t6] = 0
R15 [t7] = 0
R16 [s0] = 9
R17 [s1] = 2
R18 [s2] = 11
R19 [s3] = 0
R20 [s4] = 0
R21 [s5] = 0
R22 [s6] = 0
R23 [s7] = 0
R24 [t8] = 0
R25 [t9] = 0
R26 [k0] = 0
R27 [k1] = 0
R28 [gp] = 268468224
R29 [sp] = 2147483144
R30 [s8] = 0
R31 [ra] = 4194360
```

อธิบาย :

จากคำสั่งข้างต้น จะนำค่าใน $t3 มาเทียบกับ $0 ว่าค่าใน register $t3 มีค่าเท่ากับ 0 หรือไม่ถ้าเท่ากับ 0 จะกระโดดไปทำที่ bistwise แต่ถ้าไม่เท่ากับ 0 จะทำบรรทัดถัดไป ซึ่งในที่นี้ $11 มีค่าเท่ากับ 0x0000001 ทำให้ทำบรรทัดต่อไป

```
slti $t3, $t2, 4
```

```
R0  [r0] = 0
R1  [at] = 0
R2  [v0] = 4
R3  [v1] = 0
R4  [a0] = 1
R5  [a1] = 2147483148
R6  [a2] = 2147483156
R7  [a3] = 0
R8  [t0] = 5
R9  [t1] = 7
R10 [t2] = 12
R11 [t3] = 0
R12 [t4] = 0
R13 [t5] = 0
R14 [t6] = 0
R15 [t7] = 0
R16 [s0] = 9
R17 [s1] = 2
R18 [s2] = 11
R19 [s3] = 0
R20 [s4] = 0
R21 [s5] = 0
R22 [s6] = 0
R23 [s7] = 0
R24 [t8] = 0
R25 [t9] = 0
R26 [k0] = 0
R27 [k1] = 0
R28 [gp] = 268468224
R29 [sp] = 2147483144
R30 [s8] = 0
R31 [ra] = 4194360
```

อธิบาย :

จากคำสั่งข้างต้น จะนำค่าใน $t2 มาเทียบกับค่าคงที่ 4 ว่าค่าใน register $t2 น้อยกว่าหรือไม่ ถ้าเป็นจริงผลลัพธ์จะมีค่าเท่ากับ 1 แต่ถ้าเป็นเท็จผลลัพธ์จะมีค่าเท่ากับ 0 แล้วนำผลลัพธ์ที่ได้มาใส่ที่ register $11 ดังนั้น register $11 จึงมีค่าเท่ากับ 0x0000000 ทำให้ทำบรรทัดต่อไป

นางสาวจิรัชฌา วิทยาชีวะ 6010502519
นางสาวจารวี ทองเจริญ 6010504660

```
subu $t2, $t0, $t1
```

```
R0  [r0] = 0
R1  [at] = 0
R2  [v0] = 4
R3  [v1] = 0
R4  [a0] = 1
R5  [a1] = 7ffffe0c
R6  [a2] = 7ffffe14
R7  [a3] = 0
R8  [t0] = 5
R9  [t1] = 7
R10 [t2] = fffffffe
R11 [t3] = 0
R12 [t4] = 0
R13 [t5] = 0
R14 [t6] = 0
R15 [t7] = 0
R16 [s0] = 9
R17 [s1] = 2
R18 [s2] = b
R19 [s3] = 0
R20 [s4] = 0
R21 [s5] = 0
R22 [s6] = 0
R23 [s7] = 0
R24 [t8] = 0
R25 [t9] = 0
R26 [k0] = 0
R27 [k1] = 0
R28 [gp] = 10008000
R29 [sp] = 7ffffe08
R30 [s8] = 0
R31 [ra] = 400038
```

อธิบาย :

จากคำสั่งข้างต้น จะนำค่าใน $t0 มาลบกับค่าใน $t1 (แบบ unsigned) แล้วนำผลลัพธ์มาใส่ที่ register $t2 ดังนั้น register $10 จึงเท่ากับ 0xfffffffe

```
sltu $t3, $t2, $t0
```

```
R0  [r0] = 0
R1  [at] = 0
R2  [v0] = 4
R3  [v1] = 0
R4  [a0] = 1
R5  [a1] = 7ffffe0c
R6  [a2] = 7ffffe14
R7  [a3] = 0
R8  [t0] = 5
R9  [t1] = 7
R10 [t2] = fffffffe
R11 [t3] = 0
R12 [t4] = 0
R13 [t5] = 0
R14 [t6] = 0
R15 [t7] = 0
R16 [s0] = 9
R17 [s1] = 2
R18 [s2] = b
R19 [s3] = 0
R20 [s4] = 0
R21 [s5] = 0
R22 [s6] = 0
R23 [s7] = 0
R24 [t8] = 0
R25 [t9] = 0
R26 [k0] = 0
R27 [k1] = 0
R28 [gp] = 10008000
R29 [sp] = 7ffffe08
R30 [s8] = 0
R31 [ra] = 400038
```

อธิบาย :

จากคำสั่งข้างต้น จะนำค่าใน $t2 มาเทียบกับค่าใน $t0 ว่าค่าใน register $t2 น้อยกว่าหรือไม่ (แบบ unsigned) ถ้าเป็นจริงผลลัพธ์จะมีค่าเท่ากับ 1 แต่ถ้าเป็นเท็จผลลัพธ์จะมีค่าเท่ากับ 0 แล้วนำผลลัพธ์ที่ได้มาใส่ที่ register $t3 ดังนั้น register $11 จึงมีค่าเท่ากับ 0x0000000

นางสาวจิรัชฌา วิทยาชีวะ 6010502519
นางสาวจารวี ทองเจริญ 6010504660

```
bne $t3, $zero, bistwise
```

อธิบาย :

จากคำสั่งข้างต้น จะนำค่าใน $t3 มาเทียบกับ $0 ว่าค่าใน register $t3 มีค่าไม่เท่ากับ 0 หรือไม่ถ้าไม่เท่ากับ 0 จะกระโดดไปทำที่ bistwise แต่ถ้าเท่ากับ 0 จะทำบรรทัดถัดไป ซึ่งในที่นี้ $11 มีค่าเท่ากับ 0x0000000

```
R0  [r0] = 0
R1  [at] = 0
R2  [v0] = 4
R3  [v1] = 0
R4  [a0] = 1
R5  [a1] = 7ffffe0c
R6  [a2] = 7ffffe14
R7  [a3] = 0
R8  [t0] = 5
R9  [t1] = 7
R10 [t2] = fffffffe
R11 [t3] = 0
R12 [t4] = 0
R13 [t5] = 0
R14 [t6] = 0
R15 [t7] = 0
R16 [s0] = 9
R17 [s1] = 2
R18 [s2] = b
R19 [s3] = 0
R20 [s4] = 0
R21 [s5] = 0
R22 [s6] = 0
R23 [s7] = 0
R24 [t8] = 0
R25 [t9] = 0
R26 [k0] = 0
R27 [k1] = 0
R28 [gp] = 10008000
R29 [sp] = 7ffffe08
R30 [s8] = 0
R31 [ra] = 400038
```

```
slt $t3, $t0, $t1
```

อธิบาย :

จากคำสั่งข้างต้น จะนำค่าใน $t0 มาเทียบกับค่าใน $t1 ว่าค่าใน register $t0 น้อยกว่าหรือไม่ ถ้าเป็นจริงผลลัพธ์จะมีค่าเท่ากับ 1 แต่ถ้าเป็นเท็จผลลัพธ์จะมีค่าเท่ากับ 0 แล้วนำผลลัพธ์ที่ได้มาใส่ที่ register $t3 ดังนั้น register $11 จึงมีค่าเท่ากับ 0x0000001

```
R0  [r0] = 0
R1  [at] = 0
R2  [v0] = 4
R3  [v1] = 0
R4  [a0] = 1
R5  [a1] = 7ffffe0c
R6  [a2] = 7ffffe14
R7  [a3] = 0
R8  [t0] = 5
R9  [t1] = 7
R10 [t2] = fffffffe
R11 [t3] = 1
R12 [t4] = 0
R13 [t5] = 0
R14 [t6] = 0
R15 [t7] = 0
R16 [s0] = 9
R17 [s1] = 2
R18 [s2] = b
R19 [s3] = 0
R20 [s4] = 0
R21 [s5] = 0
R22 [s6] = 0
R23 [s7] = 0
R24 [t8] = 0
R25 [t9] = 0
R26 [k0] = 0
R27 [k1] = 0
R28 [gp] = 10008000
R29 [sp] = 7ffffe08
R30 [s8] = 0
R31 [ra] = 400038
```

นางสาวจิรัชฌา วิทยาชีวะ 6010502519
นางสาวจารวี ทองเจริญ 6010504660

```
bne $t3, $zero, bistwise
```

```
R0  [r0] = 0
R1  [at] = 0
R2  [v0] = 4
R3  [v1] = 0
R4  [a0] = 1
R5  [a1] = 7ffffe0c
R6  [a2] = 7ffffe14
R7  [a3] = 0
R8  [t0] = 5
R9  [t1] = 7
R10 [t2] = fffffffe
R11 [t3] = 1
R12 [t4] = 0
R13 [t5] = 0
R14 [t6] = 0
R15 [t7] = 0
R16 [s0] = 9
R17 [s1] = 2
R18 [s2] = b
R19 [s3] = 0
R20 [s4] = 0
R21 [s5] = 0
R22 [s6] = 0
R23 [s7] = 0
R24 [t8] = 0
R25 [t9] = 0
R26 [k0] = 0
R27 [k1] = 0
R28 [gp] = 10008000
R29 [sp] = 7ffffe08
R30 [s8] = 0
R31 [ra] = 400038
```

อธิบาย :

จากคำสั่งข้างต้น จะนำค่าใน $t3 มาเทียบกับ $0 ว่าค่าใน register $t3 มีค่าไม่เท่ากับ 0 หรือไม่ถ้าไม่เท่ากับ 0 จะกระโดดไปทำที่ bistwise แต่ถ้าเท่ากับ 0 จะทำบรรทัดถัดไป ซึ่งในที่นี้ $11 มีค่าเท่ากับ 0x00000001

Bistwise:

```
and $t4, $t0, $t1
```

```
R0  [r0] = 0
R1  [at] = 0
R2  [v0] = 4
R3  [v1] = 0
R4  [a0] = 1
R5  [a1] = 7ffffe0c
R6  [a2] = 7ffffe14
R7  [a3] = 0
R8  [t0] = 5
R9  [t1] = 7
R10 [t2] = fffffffe
R11 [t3] = 1
R12 [t4] = 5
R13 [t5] = 0
R14 [t6] = 0
R15 [t7] = 0
R16 [s0] = 9
R17 [s1] = 2
R18 [s2] = b
R19 [s3] = 0
R20 [s4] = 0
R21 [s5] = 0
R22 [s6] = 0
R23 [s7] = 0
R24 [t8] = 0
R25 [t9] = 0
R26 [k0] = 0
R27 [k1] = 0
R28 [gp] = 0
R29 [sp] = 7ffffe08
R30 [s8] = 0
R31 [ra] = 400038
```

อธิบาย :

จากคำสั่งข้างต้น จะนำค่าใน $t0 และ $t1 มา AND กัน และนำค่าไปใส่ไว้ที่ $t4 ดังนั้น Register $12 มีค่าเท่ากับ 0x00000005

นางสาวจิรัชฌา วิทยาชีวะ 6010502519
นางสาวจารวี ทองเจริญ 6010504660

```
or $t5, $t0, $t1
```

อธิบาย :

จากคำสั่งข้างต้น จะนำค่าใน $t0 และ $t1 มา OR กัน และนำค่าไปใส่ไว้ที่ $t3 ดังนั้น Register $12 มีค่าเท่ากับ 0x00000007

```
R0  [r0] = 0
R1  [at] = 0
R2  [v0] = 4
R3  [v1] = 0
R4  [a0] = 1
R5  [a1] = 7ffffe0c
R6  [a2] = 7ffffe14
R7  [a3] = 0
R8  [t0] = 5
R9  [t1] = 7
R10 [t2] = fffffffe
R11 [t3] = 1
R12 [t4] = 5
R13 [t5] = 7
R14 [t6] = 0
R15 [t7] = 0
R16 [s0] = 9
R17 [s1] = 2
R18 [s2] = b
R19 [s3] = 0
R20 [s4] = 0
R21 [s5] = 0
R22 [s6] = 0
R23 [s7] = 0
R24 [t8] = 0
R25 [t9] = 0
R26 [k0] = 0
R27 [k1] = 0
R28 [gp] = 0
R29 [sp] = 7ffffe08
R30 [s8] = 0
R31 [ra] = 400038
```

```
xor $t6, $t0, $t1
```

อธิบาย :

จากคำสั่งข้างต้น จะนำค่าใน $t0 และ $t1 มา XOR กัน และนำค่าไปใส่ไว้ที่ $t6 ดังนั้น Register $14 มีค่าเท่ากับ 0x00000002

```
R0  [r0] = 0
R1  [at] = 0
R2  [v0] = 4
R3  [v1] = 0
R4  [a0] = 1
R5  [a1] = 7ffffe0c
R6  [a2] = 7ffffe14
R7  [a3] = 0
R8  [t0] = 5
R9  [t1] = 7
R10 [t2] = fffffffe
R11 [t3] = 1
R12 [t4] = 5
R13 [t5] = 7
R14 [t6] = 2
R15 [t7] = 0
R16 [s0] = 9
R17 [s1] = 2
R18 [s2] = b
R19 [s3] = 0
R20 [s4] = 0
R21 [s5] = 0
R22 [s6] = 0
R23 [s7] = 0
R24 [t8] = 0
R25 [t9] = 0
R26 [k0] = 0
R27 [k1] = 0
R28 [gp] = 0
R29 [sp] = 7ffffe08
R30 [s8] = 0
R31 [ra] = 400038
```

นางสาวจิรัชฌา วิทยาชีวะ 6010502519
นางสาวจารวี ทองเจริญ 6010504660

```
nor $t7, $t0, $t1
```

```
R0  [r0] = 0
R1  [at] = 0
R2  [v0] = 4
R3  [v1] = 0
R4  [a0] = 1
R5  [a1] = 7ffffe0c
R6  [a2] = 7ffffe14
R7  [a3] = 0
R8  [t0] = 5
R9  [t1] = 7
R10 [t2] = fffffffe
R11 [t3] = 1
R12 [t4] = 5
R13 [t5] = 7
R14 [t6] = 2
R15 [t7] = fffffff8
R16 [s0] = 9
R17 [s1] = 2
R18 [s2] = b
R19 [s3] = 0
R20 [s4] = 0
R21 [s5] = 0
R22 [s6] = 0
R23 [s7] = 0
R24 [t8] = 0
R25 [t9] = 0
R26 [k0] = 0
R27 [k1] = 0
R28 [gp] = 0
R29 [sp] = 7ffffe08
R30 [s8] = 0
R31 [ra] = 400038
```

อธิบาย :

จากคำสั่งข้างต้น จะนำค่าใน $t0 และ $t1 มา NOR กัน และนำค่าไปใส่ไว้ที่ $t7 ดังนั้น Register $15 มีค่าเท่ากับ fxfffff8

```
j load_store
```

```
R0  [r0] = 0
R1  [at] = 0
R2  [v0] = 4
R3  [v1] = 0
R4  [a0] = 1
R5  [a1] = 7ffffe0c
R6  [a2] = 7ffffe14
R7  [a3] = 0
R8  [t0] = 5
R9  [t1] = 7
R10 [t2] = fffffffe
R11 [t3] = 1
R12 [t4] = 5
R13 [t5] = 7
R14 [t6] = 2
R15 [t7] = fffffff8
R16 [s0] = 9
R17 [s1] = 2
R18 [s2] = b
R19 [s3] = 0
R20 [s4] = 0
R21 [s5] = 0
R22 [s6] = 0
R23 [s7] = 0
R24 [t8] = 0
R25 [t9] = 0
R26 [k0] = 0
R27 [k1] = 0
R28 [gp] = 0
R29 [sp] = 7ffffe08
R30 [s8] = 0
R31 [ra] = 400038
```

อธิบาย :

จากคำสั่งข้างต้น จะโดดไปยังคำสั่ง load_store แต่ไม่ได้เก็บ address ไว้ที่ $ra ดังนั้น register $31 จึงไม่เปลี่ยนแปลง ดังรูป

นางสาวจิรัชฌา วิทยาชีวะ 6010502519
นางสาวจารวี ทองเจริญ 6010504660

Load_store:

```
addiu $sp, $sp, -4
```

อธิบาย :

จากคำสั่งข้างต้นจะทำการนำค่า $sp ซึ่งมีค่าเท่ากับ 7ffffe08 มาบวกกับ -4 (แบบ unsigned) แล้วมาใส่ใน $sp เหมือน ดังนั้นค่า Register $29 จึงมีค่าเท่ากับ 7ffffe04

```
R0  [r0] = 0
R1  [at] = 0
R2  [v0] = 4
R3  [v1] = 0
R4  [a0] = 1
R5  [a1] = 7ffffe0c
R6  [a2] = 7ffffe14
R7  [a3] = 0
R8  [t0] = 5
R9  [t1] = 7
R10 [t2] = fffffffe
R11 [t3] = 1
R12 [t4] = 5
R13 [t5] = 7
R14 [t6] = 2
R15 [t7] = fffffff8
R16 [s0] = 9
R17 [s1] = 2
R18 [s2] = b
R19 [s3] = 0
R20 [s4] = 0
R21 [s5] = 0
R22 [s6] = 0
R23 [s7] = 0
R24 [t8] = 0
R25 [t9] = 0
R26 [k0] = 0
R27 [k1] = 0
R28 [gp] = 0
R29 [sp] = 7ffffe04
R30 [s8] = 0
R31 [ra] = 400038
```

```
sw $t0, 0($sp)
```

อธิบาย :

จากคำสั่งข้างต้นจะทำการ store ค่า $t0 ไว้ที่ 0($sp) ซึ่งเราทราบ แล้วว่า $sp มีค่าเท่ากับ 7ffffe04 ดังนั้นจึงเก็บค่า 0x00000005 ไว้ที่ 7ffffe04

```
R0  [r0] = 0
R1  [at] = 0
R2  [v0] = 4
R3  [v1] = 0
R4  [a0] = 1
R5  [a1] = 7ffffe0c
R6  [a2] = 7ffffe14
R7  [a3] = 0
R8  [t0] = 5
R9  [t1] = 7
R10 [t2] = fffffffe
R11 [t3] = 1
R12 [t4] = 5
R13 [t5] = 7
R14 [t6] = 2
R15 [t7] = fffffff8
R16 [s0] = 9
R17 [s1] = 2
R18 [s2] = b
R19 [s3] = 0
R20 [s4] = 0
R21 [s5] = 0
R22 [s6] = 0
R23 [s7] = 0
R24 [t8] = 0
R25 [t9] = 0
R26 [k0] = 0
R27 [k1] = 0
R28 [gp] = 0
R29 [sp] = 7ffffe04
R30 [s8] = 0
R31 [ra] = 400038
```

นางสาวจิรัชฌา วิทยาชีวะ 6010502519
นางสาวจารวี ทองเจริญ 6010504660

```
addiu $sp, $sp, -4
```

อธิบาย :

จากคำสั่งข้างต้นจะทำการนำค่า $sp ซึ่งมีค่าเท่ากับ 7ffffe04 มาบวกกับ -4 (แบบ unsigned) แล้วมาใส่ใน $sp เหมือน ดังนั้นค่า Register $29 จึงมีค่า เท่ากับ 7ffffe00

```
R0  [r0] = 0
R1  [at] = 0
R2  [v0] = 4
R3  [v1] = 0
R4  [a0] = 1
R5  [a1] = 7ffffe0c
R6  [a2] = 7ffffe14
R7  [a3] = 0
R8  [t0] = 5
R9  [t1] = 7
R10 [t2] = fffffffe
R11 [t3] = 1
R12 [t4] = 5
R13 [t5] = 7
R14 [t6] = 2
R15 [t7] = fffffff8
R16 [s0] = 9
R17 [s1] = 2
R18 [s2] = b
R19 [s3] = 0
R20 [s4] = 0
R21 [s5] = 0
R22 [s6] = 0
R23 [s7] = 0
R24 [t8] = 0
R25 [t9] = 0
R26 [k0] = 0
R27 [k1] = 0
R28 [gp] = 0
R29 [sp] = 7ffffe00
R30 [s8] = 0
R31 [ra] = 400038
```

```
sw $t1, 0($sp)
```

อธิบาย :

จากคำสั่งข้างต้นจะทำการ store ค่า $t1 ไว้ที่ 0($sp) ซึ่งเราทราบ แล้วว่า $sp มีค่าเท่ากับ 7ffffe00 ดังนั้นจึงเก็บค่า 0x00000007 ไว้ที่ 7ffffe00

```
R0  [r0] = 0
R1  [at] = 0
R2  [v0] = 4
R3  [v1] = 0
R4  [a0] = 1
R5  [a1] = 7ffffe0c
R6  [a2] = 7ffffe14
R7  [a3] = 0
R8  [t0] = 5
R9  [t1] = 7
R10 [t2] = fffffffe
R11 [t3] = 1
R12 [t4] = 5
R13 [t5] = 7
R14 [t6] = 2
R15 [t7] = fffffff8
R16 [s0] = 9
R17 [s1] = 2
R18 [s2] = b
R19 [s3] = 0
R20 [s4] = 0
R21 [s5] = 0
R22 [s6] = 0
R23 [s7] = 0
R24 [t8] = 0
R25 [t9] = 0
R26 [k0] = 0
R27 [k1] = 0
R28 [gp] = 0
R29 [sp] = 7ffffe00
R30 [s8] = 0
R31 [ra] = 400038
```

นางสาวจิรัชฌา วิทยาชีวะ 6010502519
นางสาวจารวี ทองเจริญ 6010504660

```
lw $t3, 4($sp)
```

```
R0  [r0] = 0
R1  [at] = 0
R2  [v0] = 4
R3  [v1] = 0
R4  [a0] = 1
R5  [a1] = 7fffe0c
R6  [a2] = 7fffe14
R7  [a3] = 0
R8  [t0] = 5
R9  [t1] = 7
R10 [t2] = fffffffe
R11 [t3] = 5
R12 [t4] = 5
R13 [t5] = 7
R14 [t6] = 2
R15 [t7] = ffffff8
R16 [s0] = 9
R17 [s1] = 2
R18 [s2] = b
R19 [s3] = 0
R20 [s4] = 0
R21 [s5] = 0
R22 [s6] = 0
R23 [s7] = 0
R24 [t8] = 0
R25 [t9] = 0
R26 [k0] = 0
R27 [k1] = 0
R28 [gp] = 0
R29 [sp] = 7fffe00
R30 [s8] = 0
R31 [ra] = 400038
```

อธิบาย :

จากคำสั่งข้างต้นจะทำการ load ค่าจาก 4($sp) มาซึ่ง $sp มีค่าเท่ากับ 7ffffe00 ดังนั้น จะทำการนำค่าจาก stack ที่ $sp มีค่าเท่ากับ 7ffffe04 มาไว้ที่ $t3 ดังนั้น Register $11 มีค่าเท่ากับ 0x00000005

```
lw $t4, 0($sp)
```

```
R0  [r0] = 0
R1  [at] = 0
R2  [v0] = 4
R3  [v1] = 0
R4  [a0] = 1
R5  [a1] = 7fffe0c
R6  [a2] = 7fffe14
R7  [a3] = 0
R8  [t0] = 5
R9  [t1] = 7
R10 [t2] = fffffffe
R11 [t3] = 5
R12 [t4] = 7
R13 [t5] = 7
R14 [t6] = 2
R15 [t7] = ffffff8
R16 [s0] = 9
R17 [s1] = 2
R18 [s2] = b
R19 [s3] = 0
R20 [s4] = 0
R21 [s5] = 0
R22 [s6] = 0
R23 [s7] = 0
R24 [t8] = 0
R25 [t9] = 0
R26 [k0] = 0
R27 [k1] = 0
R28 [gp] = 0
R29 [sp] = 7fffe00
R30 [s8] = 0
R31 [ra] = 400038
```

อธิบาย :

จากคำสั่งข้างต้นจะทำการ load ค่าจาก 0($sp) มาซึ่ง $sp มีค่าเท่ากับ 7ffffe00 ดังนั้น จะทำการนำค่าจาก stack ที่ $sp มีค่าเท่ากับ 7ffffe00 มาไว้ที่ $t4 ดังนั้น Register $12 มีค่าเท่ากับ 0x00000007

นางสาวจิรัชฌา วิทยาชีวะ 6010502519
นางสาวจารวี ทองเจริญ 6010504660

```
addiu $sp, $sp, 8
```

อธิบาย :

จากคำสั่งข้างต้นจะทำการนำค่า $sp ซึ่งมีค่าเท่ากับ7ffffe00 มาบวกกับ 8 (แบบ unsigned) แล้วมาใส่ใน $sp เหมือนเดิม ดังนั้นค่า Register $29 จึงมีค่า เท่ากับ 7ffffe08

```
R0  [r0] = 0
R1  [at] = 0
R2  [v0] = 4
R3  [v1] = 0
R4  [a0] = 1
R5  [a1] = 7ffffe0c
R6  [a2] = 7ffffe14
R7  [a3] = 0
R8  [t0] = 5
R9  [t1] = 7
R10 [t2] = fffffffe
R11 [t3] = 5
R12 [t4] = 7
R13 [t5] = 7
R14 [t6] = 2
R15 [t7] = fffffff8
R16 [s0] = 9
R17 [s1] = 2
R18 [s2] = b
R19 [s3] = 0
R20 [s4] = 0
R21 [s5] = 0
R22 [s6] = 0
R23 [s7] = 0
R24 [t8] = 0
R25 [t9] = 0
R26 [k0] = 0
R27 [k1] = 0
R28 [gp] = 0
R29 [sp] = 7ffffe08
R30 [s8] = 0
R31 [ra] = 400038
```

```
j bistwise_2
```

อธิบาย :

จากคำสั่งข้างต้น จะโดดไปยังคำสั่ง bistwise_2l แต่ไม่ได้เก็บ address ไว้ที่ $ra ดังนั้น register $31 จึงไม่เปลี่ยนแปลง ดังรูป

```
R0  [r0] = 0
R1  [at] = 0
R2  [v0] = 4
R3  [v1] = 0
R4  [a0] = 1
R5  [a1] = 7ffffe0c
R6  [a2] = 7ffffe14
R7  [a3] = 0
R8  [t0] = 5
R9  [t1] = 7
R10 [t2] = fffffffe
R11 [t3] = 5
R12 [t4] = 7
R13 [t5] = 7
R14 [t6] = 2
R15 [t7] = fffffff8
R16 [s0] = 9
R17 [s1] = 2
R18 [s2] = b
R19 [s3] = 0
R20 [s4] = 0
R21 [s5] = 0
R22 [s6] = 0
R23 [s7] = 0
R24 [t8] = 0
R25 [t9] = 0
R26 [k0] = 0
R27 [k1] = 0
R28 [gp] = 0
R29 [sp] = 7ffffe08
R30 [s8] = 0
R31 [ra] = 400038
```

นางสาวจิรัชฌา วิทยาชีวะ 6010502519
นางสาวจารวี ทองเจริญ 6010504660

Bistwise_2:

```
andi $t4, $t0, 12
```

อธิบาย :

จากคำสั่งข้างต้น จะนำค่าใน $t0 มา AND กับค่าคงที่ 12 และนำค่าไปใส่ไว้ที่ $t4 ดังนั้น Register $12 มีค่าเท่ากับ 0x00000004

```
R0  [r0] = 0
R1  [at] = 0
R2  [v0] = 4
R3  [v1] = 0
R4  [a0] = 1
R5  [a1] = 7ffffe0c
R6  [a2] = 7ffffe14
R7  [a3] = 0
R8  [t0] = 5
R9  [t1] = 7
R10 [t2] = fffffffe
R11 [t3] = 5
R12 [t4] = 4
R13 [t5] = 7
R14 [t6] = 2
R15 [t7] = fffffff8
R16 [s0] = 9
R17 [s1] = 2
R18 [s2] = b
R19 [s3] = 0
R20 [s4] = 0
R21 [s5] = 0
R22 [s6] = 0
R23 [s7] = 0
R24 [t8] = 0
R25 [t9] = 0
R26 [k0] = 0
R27 [k1] = 0
R28 [gp] = 0
R29 [sp] = 7ffffe08
R30 [s8] = 0
R31 [ra] = 400038
```

```
ori $t5, $t0, 13
```

อธิบาย :

จากคำสั่งข้างต้น จะนำค่าใน $t0 และ ค่าคงที่ 13 มา OR กัน และนำค่าไปใส่ไว้ที่ $t5 ดังนั้น Register $13 มีค่าเท่ากับ 0x0000000d

```
R0  [r0] = 0
R1  [at] = 0
R2  [v0] = 4
R3  [v1] = 0
R4  [a0] = 1
R5  [a1] = 7ffffe0c
R6  [a2] = 7ffffe14
R7  [a3] = 0
R8  [t0] = 5
R9  [t1] = 7
R10 [t2] = fffffffe
R11 [t3] = 5
R12 [t4] = 4
R13 [t5] = d
R14 [t6] = 2
R15 [t7] = fffffff8
R16 [s0] = 9
R17 [s1] = 2
R18 [s2] = b
R19 [s3] = 0
R20 [s4] = 0
R21 [s5] = 0
R22 [s6] = 0
R23 [s7] = 0
R24 [t8] = 0
R25 [t9] = 0
R26 [k0] = 0
R27 [k1] = 0
R28 [gp] = 0
R29 [sp] = 7ffffe08
R30 [s8] = 0
R31 [ra] = 400038
```

นางสาวจิรัชฌา วิทยาชีวะ 6010502519
นางสาวจารวี ทองเจริญ 6010504660

```
xori $t6, $t0, 14
```

อธิบาย :

จากคำสั่งข้างต้น จะนำค่าใน $t0 และ ค่าคงที่ 14 มา XOR กัน และนำค่าไปใส่ไว้ที่ $t6 ดังนั้น Register $14 มีค่าเท่ากับ 0x0000000b

```
R0  [r0] = 0
R1  [at] = 0
R2  [v0] = 4
R3  [v1] = 0
R4  [a0] = 1
R5  [a1] = 7ffffe0c
R6  [a2] = 7ffffe14
R7  [a3] = 0
R8  [t0] = 5
R9  [t1] = 7
R10 [t2] = fffffffe
R11 [t3] = 5
R12 [t4] = 4
R13 [t5] = d
R14 [t6] = b
R15 [t7] = fffffff8
R16 [s0] = 9
R17 [s1] = 2
R18 [s2] = b
R19 [s3] = 0
R20 [s4] = 0
R21 [s5] = 0
R22 [s6] = 0
R23 [s7] = 0
R24 [t8] = 0
R25 [t9] = 0
R26 [k0] = 0
R27 [k1] = 0
R28 [gp] = 0
R29 [sp] = 7ffffe08
R30 [s8] = 0
R31 [ra] = 400038
```

```
jr $ra
```

อธิบาย :

จากคำสั่งข้างต้น จะมีการกระโดด ไปที่โปรแกรมทำสุดท้ายก่อนที่จะทำการ jal โดยดู addreass จาก $ra หรือ Register $31 ที่ตอนนี้มีค่า 0X400038

```
R0  [r0] = 0
R1  [at] = 0
R2  [v0] = 4
R3  [v1] = 0
R4  [a0] = 1
R5  [a1] = 7ffffe0c
R6  [a2] = 7ffffe14
R7  [a3] = 0
R8  [t0] = 5
R9  [t1] = 7
R10 [t2] = fffffffe
R11 [t3] = 5
R12 [t4] = 4
R13 [t5] = d
R14 [t6] = b
R15 [t7] = fffffff8
R16 [s0] = 9
R17 [s1] = 2
R18 [s2] = b
R19 [s3] = 0
R20 [s4] = 0
R21 [s5] = 0
R22 [s6] = 0
R23 [s7] = 0
R24 [t8] = 0
R25 [t9] = 0
R26 [k0] = 0
R27 [k1] = 0
R28 [gp] = 0
R29 [sp] = 7ffffe08
R30 [s8] = 0
R31 [ra] = 400038
```

นางสาวจิรัชฌา วิทยาชีวะ 6010502519
นางสาวจารวี ทองเจริญ 6010504660

Jal in Main:

```
sll $t2, $t0, 1
```

อธิบาย :

จากคำสั่งข้างต้น จะนำค่าจาก register $t0 ซึ่งมีค่า เท่ากับ 0x00000005 มา shift left 1 ครั้ง แล้วนำค่าที่ ได้มาใส่ที่ $t2 ดังนั้นที่ register $10 ถึงมีค่าเท่ากับ 0x0000000a

```
R0  [r0] = 0
R1  [at] = 0
R2  [v0] = 4
R3  [v1] = 0
R4  [a0] = 1
R5  [a1] = 7ffffe0c
R6  [a2] = 7ffffe14
R7  [a3] = 0
R8  [t0] = 5
R9  [t1] = 7
R10 [t2] = a
R11 [t3] = 5
R12 [t4] = 4
R13 [t5] = d
R14 [t6] = b
R15 [t7] = fffffff8
R16 [s0] = 9
R17 [s1] = 2
R18 [s2] = b
R19 [s3] = 0
R20 [s4] = 0
R21 [s5] = 0
R22 [s6] = 0
R23 [s7] = 0
R24 [t8] = 0
R25 [t9] = 0
R26 [k0] = 0
R27 [k1] = 0
R28 [gp] = 0
R29 [sp] = 7ffffe08
R30 [s8] = 0
R31 [ra] = 400038
```

นางสาวจิรัชฌา วิทยาชีวะ 6010502519
นางสาวจารวี ทองเจริญ 6010504660

```
srl $t2, $t0, 1
```

อธิบาย :

จากคำสั่งข้างต้น จะนำค่าจาก register $t0 ซึ่งมีค่า เท่ากับ 0x00000005 มา shift right 1 ครั้ง แล้วนำค่าที่ ได้มาใส่ที่ $t2 ดังนั้นที่ register $10 ถึงมีค่าเท่ากับ 0x00000002

```
R0  [r0] = 0
R1  [at] = 0
R2  [v0] = 4
R3  [v1] = 0
R4  [a0] = 1
R5  [a1] = 7ffffe0c
R6  [a2] = 7ffffe14
R7  [a3] = 0
R8  [t0] = 5
R9  [t1] = 7
R10 [t2] = 2
R11 [t3] = 5
R12 [t4] = 4
R13 [t5] = d
R14 [t6] = b
R15 [t7] = ffffffff8
R16 [s0] = 9
R17 [s1] = 2
R18 [s2] = b
R19 [s3] = 0
R20 [s4] = 0
R21 [s5] = 0
R22 [s6] = 0
R23 [s7] = 0
R24 [t8] = 0
R25 [t9] = 0
R26 [k0] = 0
R27 [k1] = 0
R28 [gp] = 0
R29 [sp] = 7ffffe08
R30 [s8] = 0
R31 [ra] = 400038
```

```
sra $t2, $t0, 1
```

อธิบาย :

จากคำสั่งข้างต้น จะนำค่าจาก register $t0 ซึ่งมีค่า เท่ากับ 0x00000005 มา shift left Arithmetic 1 ครั้ง แล้วนำค่าที่ ได้มาใส่ที่ $t2 ดังนั้นที่ register $10 ถึงมีค่าเท่ากับ 0x00000002

```
R0  [r0] = 0
R1  [at] = 0
R2  [v0] = 4
R3  [v1] = 0
R4  [a0] = 1
R5  [a1] = 7ffffe0c
R6  [a2] = 7ffffe14
R7  [a3] = 0
R8  [t0] = 5
R9  [t1] = 7
R10 [t2] = 2
R11 [t3] = 5
R12 [t4] = 4
R13 [t5] = d
R14 [t6] = b
R15 [t7] = ffffffff8
R16 [s0] = 9
R17 [s1] = 2
R18 [s2] = b
R19 [s3] = 0
R20 [s4] = 0
R21 [s5] = 0
R22 [s6] = 0
R23 [s7] = 0
R24 [t8] = 0
R25 [t9] = 0
R26 [k0] = 0
R27 [k1] = 0
R28 [gp] = 0
R29 [sp] = 7ffffe08
R30 [s8] = 0
R31 [ra] = 400038
```