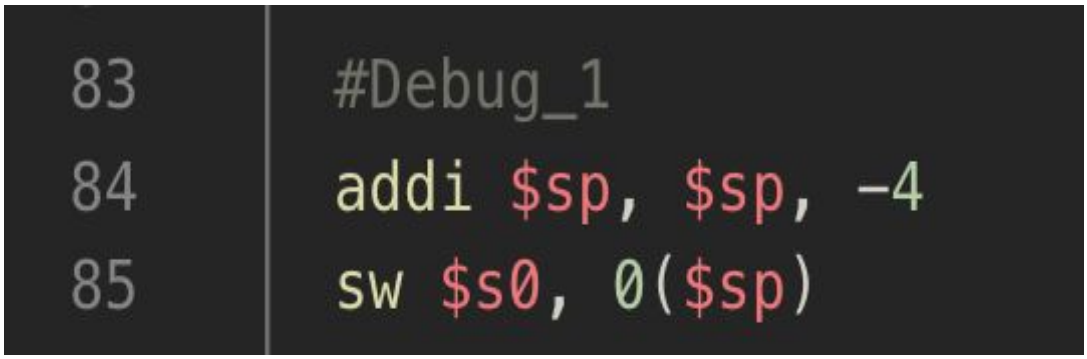
นางสาวจิรัชฌา วิทยาชีวะ 6010502519
นางสาวจารวี ทองเจริญ 6010504660

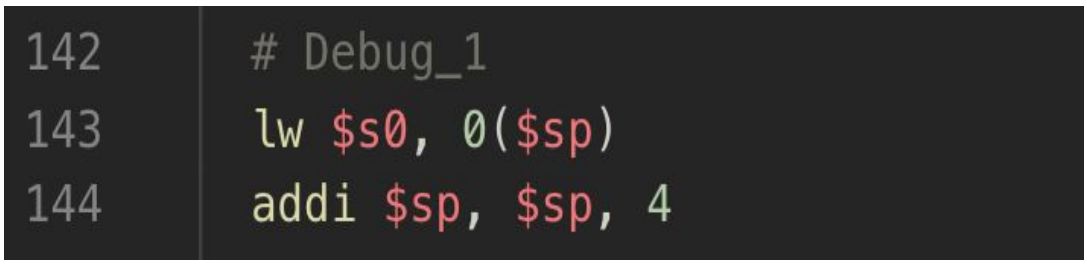
Pre_Lab2

Bug_1 : ในส่วนของ Bug ที่ 1 เป็น Bug ที่เกิดขึ้นโดยฝั่ง Caller ได้มีการประกาศใช้ Reigister $s0 ซึ่งเป็น Register ที่ฝั่ง Caller ไม่ได้ทำการ Save ใน Stack ดังนั้นเมื่อ Call Function จากโค้ดในฝั่ง Callee ซึ่งไม่ได้มีการประกาศ Register $s0 ใน Stack เช่นกัน จึงทำให้ฝั่ง Callee ได้ไปยุ่งกับค่าใน Register $s0 ในฝั่ง Caller ทำให้ค่าใน Register $s0 นั้นมีค่าที่เปลี่ยนไป

Debug_1 : ประกาศเก็บค่า Register $s0 ลงใน Stack ในฝั่ง Callee เพื่อทำให้ค่าใน Register $s0 เป็นอิสระจาก ฝั่ง Caller ซึ่งสามารถ Debug ได้ดังภาพ



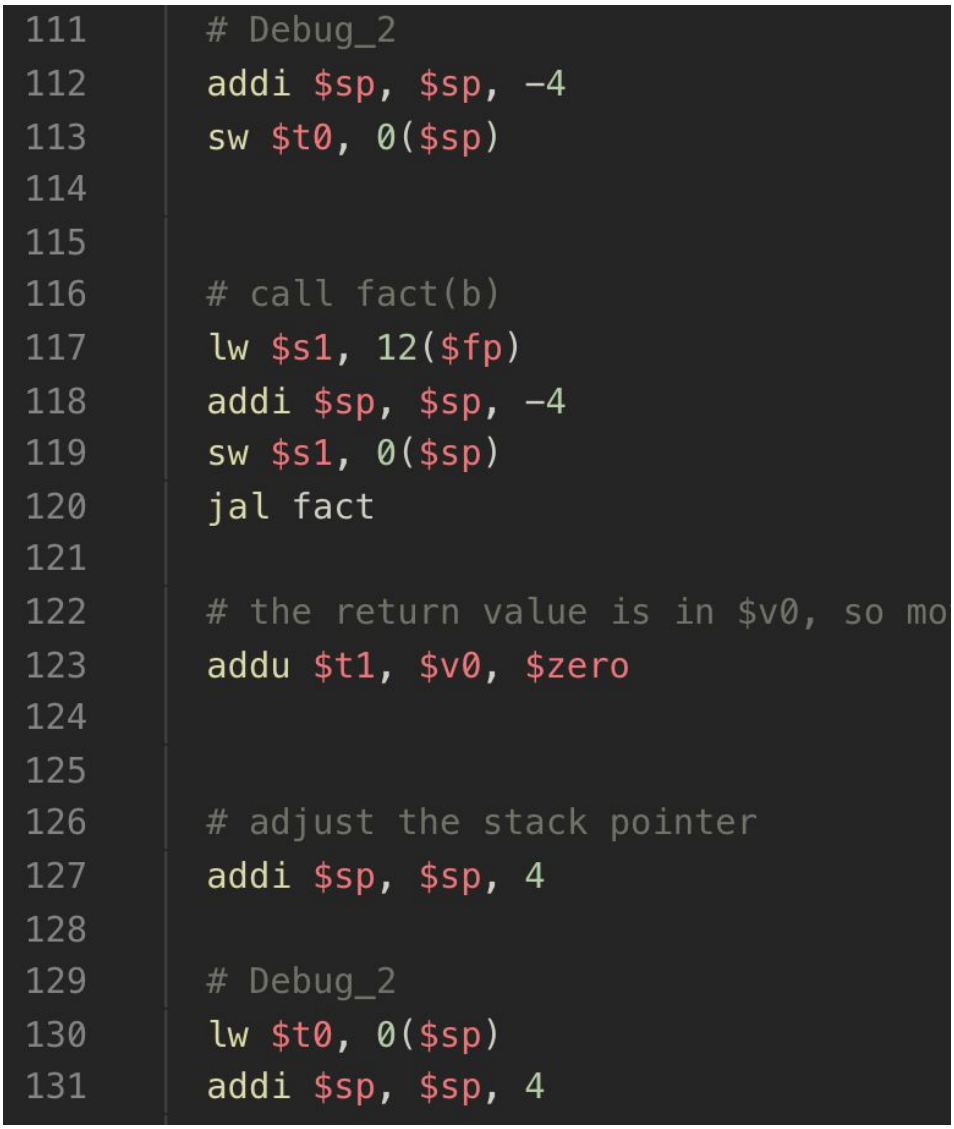
ทำการคืนค่าใน Register $s0 ซึ่งสามารถ Debug ได้ดังภาพ



นางสาวจิรัชฌา วิทยาชีวะ 6010502519
นางสาวจารวี ทองเจริญ 6010504660

Bug_2 : ในส่วนของ Bug ที่ 2 เป็น Bug ที่เกิดขึ้นโดย Register $t0 ใน Function sum_two_fact หลังจากได้ Call Function fact (เพื่อทำการคำนวณ 5!) นั้นไม่ได้มีการประกาศเก็บค่า Register $t0 ใน Stack พอเมื่อมีการ Call Function Fact อีกครั้ง (เพื่อทำการคำนวณ 3!) ซึ่งโค้ดในส่วน Function fact ได้ใช้ Register $t0 จึงทำให้มีการไปยุ่งกับค่าของ Register $t0 ซึ่งทำให้ค่าใน Register $t0 นั้นมีค่าที่เปลี่ยนไป

Debug_2 : ประกาศเก็บค่า Register $t0 ในฝั่ง Function sum_two_fact เพื่อ Save ค่าใน Register $t0 ไม่ให้เปลี่ยนแปลงแล้วจึงสามารถ Call Function Fact เพื่อคำนวณ 3! ได้โดยไม่ต้องกังวลว่าค่าใน Register $t0 จะเปลี่ยนแปลง จึงสามารถ Debug ได้ดังภาพ



```
111    # Debug_2
112    addi $sp, $sp, -4
113    sw $t0, 0($sp)
114
115
116    # call fact(b)
117    lw $s1, 12($fp)
118    addi $sp, $sp, -4
119    sw $s1, 0($sp)
120    jal fact
121
122    # the return value is in $v0, so mo
123    addu $t1, $v0, $zero
124
125
126    # adjust the stack pointer
127    addi $sp, $sp, 4
128
129    # Debug_2
130    lw $t0, 0($sp)
131    addi $sp, $sp, 4
```