Computer Architecture and Programming Languages
Jacobs University Bremen
Arsenij Percov

Course: CO20-320241
November 3rd, 2019

Homework 7

Problem 7.1

Solution:

```
my_function:
    addi $t0, $0, 10 \\add 10 to temp register
    slt $t1, $t0, $a0 \\ compare, if $a0 (x) is bigger than 10, then set $t1 to 1
    beq $t1, $0, ELSE \\ if 1 continue, else jump to else
    sub $vo, $a0, $a1\\subtract y from x, and save it to return register
    j AFTER\\jump to after
ELSE:
    add $vo, $a1, $a0\\add x and y and save them in case of else
AFTER:
    jr $ra\\return to caller
```

Problem 7.2

Solution:

```
prod:
    mult $a0, $a1 \\multiply argument 0 and 1 (x and y)
    mflo $v0 \\assuming that the result can be fit in 32 bits save to return register
    jr $ra \\ return to caller

is_more_than_fifty:
    addi $sp, $sp, -12 \\push 3 elements in stack: $ra, $a0, $a1 (they will be passed
    sw $ra, 8($sp)\\save $ra to the stack
    sw $a0, 4($sp)\\save $a0 to the stack
    sw $a1, 0($sp)\\save $a1 to the stack
    jal prod \\ call the function prod, we will pass $a0, $a1 as this call recieved t
    lw $a1, 0($sp) \\restore $a1
    lw $a0, 4($sp) \\restore $a0
    lw $ra, 8($sp) \\restore $ra
    addi $sp, $sp, 12 \\pop 3 elements from the stack
    addi $t0, $0, 50 \\ save 50 to $t0
    slt $t1, $t0, $v0 \\ if 50 is less than return from prod, then $t1 is 1
    beq $t1, $0, ELSE \\jump to else, if $t1 is equal to 0
    addi $v0, $0, 1 \\ save 1 to return register
    jr $ra \\ return to the caller
ELSE:
    move $v0, $0 \\ save 0 to return register
    jr $ra\\return to caller
```

Problem 7.3

Solution:

```
int i = 0;
while (A[i] != -1){
    i++;
}
```

Problem 7.4

Solution:

PC	Machine Code	Binary Machine Code	
60000	0 0 19 9 2 0	000000 00000 10011 01001 00010 000000	60000 is the first location of the branch
60004	0 9 22 9 0 32	000000 01001 10110 01001 00000 100000	add call
60008	35 9 8 0	100011 01001 01000 0000000000000000	
60012	4 8 21 2	000100 01000 10101 0000000000000010	
60016	8 19 19 1	001000 10011 10011 0000000000000001	
60020	2 15000	000010 00000000000011101010011000	destination = PC[0 : 3] + word address
60024		Exit	

Problem 7.5

Solution:

a) $0C000000_{16} = 00001100000000000000000000000000_2$
$201326592 = 2^{26} + 2^{27}$.
$C4630000_{16} = 11000100011000110000000000000000_2$. The number is negative, invert all bits and then add one:

$11000100011000110000000000000000_2 => 00111011100111001111111111111111$
add one:
$00111011100111001111111111111111 + 1 = 00111011100111010000000000000000$

$00111011100111010000000000000000_2 = 1000144896_{10}$
Set the right sign:
$11000100011000110000000000000000_2 = -1000144896_{10}$

b) $0C000000_{16} = 00001100000000000000000000000000_2$
$201326592 = 2^{26} + 2^{27}$.

$C4630000_{16} = 11000100011000110000000000000000_2$
$2^{16} + 2^{17} + 2^{21} + 2^{22} + 2^{26} + 2^{30} + 2^{31} = 3294822400_{10}$

c)000011 = jal jal ” ” (at address 0) Second number:
110001 00011 00011 0000000000000000

LWC1 $v1 $v1 offset
110001 00011 00011 0000000000000000 =
LWC1 $v1 0 $v1