OLIVER MENSAH
COA
PRACTICAL 7 , OBJECTIVE 6-8

(1) Understand procedures

A procedure is a construct to reduce the number of times you must repeat these same

instructions in your code. In order for a procedure to be useful, it must contain instructions.

It can either return something or just contain statements with no returning statement. For

instance, below is a procedure

procedure:

```
addi $sp, $sp, -4 # $sp -= 1; /*allocate 1 word*/
sw $ra, 0($sp) # *($sp + 0) = $ra; /*save $ra*/
add $a0, $a0, $a1 # $a0 = $a0 + $a1;
lw $t0, 4($sp) # $t0 = *($sp + 1); /* Get arg on stack */
add $a0, $a0, $t0 # $a0 = $a0 + $t0;
jal procedure2 # $v0 = procedure2($a0);
addi $v0, $v0, 50 # $v0 = $v0 + 50;
lw $ra, 0($sp) # $ra = *($sp + 0); /*restore $ra*/
addi $sp, $sp, 4 # $sp += 1; /*deallocate 1 word*/
jr $ra # return $v0
```

(2) Understanding of IO operations

IO operations are used to read or print values or strings from input/output window, and

indicate program end. For instance, we use **syscall** for operating system routine call. To do

so, we first supply appropriate values in registers $v0 and $a0-$a1 and then, the result value

(if any) are returned in register $v0.

(3) Practical understanding of processor architecture

Basically, processor architecture provides the support the MIPS programming system. In a

nutshell, MIPS is a low level systems programming language and associated tools designed

to support the development of information processing applications