```
#  ALGORITHM:
#  1.   Initialize Counter to 1
#  2.   Set Counter limit to 9
#  3.   Read a number
#  4.   Initialize Maximum to that value
#  5.   Read a number
#  6.   If number <= Maximum, go to (8)
#  7.   Set Maximum to number
#  8.   Increment Counter
#  9.   If Counter <=  Limit, go to (5)
#  10. Print Maximum
#  11. Stop

#  CODE:
# Maximum: $t0, Counter: $t1, Counter Limit: $t2
.text
main:   li $t1, 1                       # Initialize Cntr
        li $t2, 9                       # Initialize Cntr-Lmt
         li $v0, 5                      # load code for "read integer"
        syscall                         # read next number (into $v0)
        move $t0, $v0                   # set Max to that value
next:   li $v0, 5                       # load code for "read integer"
        syscall                         # read next number (into $v0)
         ble $v0, $t0, cont             # check if < or  = Max
         move $t0, $v0                  # if not, update Max
cont:   addi $t1, $t1, 1                # increment counter
        ble $t1, $t2, next              # if < limit, do again
        move $a0, $t0                   # prepare argument for print
         li $v0, 1                      # load code for "print integer"
        syscall                          # will print integer in $a0
        li $v0, 10                      # load code for "stop
        syscall                         # end execution
```

```
#  ALGORITHM MODIFIED TO INCLUDE PROMPTS AND OUTPUT MESSAGE
#  1.    Initialize Counter to 1
#  2.    Set Counter limit to 9
#  3a.  Print prompt
#  3b.  Read a number
#  4.    Initialize Maximum to that value
#  5a.   Print prompt
#  5b.   Read a number
#  6.     If number <= Maximum, go to (8)
#  7.    Set Maximum to number
#  8.    Increment Counter
#  9.    If Counter <=  Limit, go to (5a)
#  10a: Print output message
#  10b. Print Maximum
#  11.   Stop

#  CODE:
# Maximum: $t0, Counter: $t1, Counter Limit: $t2
#
.data
        Prompt: .asciiz "Enter a number: "
        OutMssg: .asciiz "\n The maximum is: "
.text
main:  li $t1, 1                        # Initialize Cntr
       li $t2, 9                        # Initialize Cntr-Lmt
       la $a0, Prompt                   # load prompt address
        li $v0, 4                       # load syscode for "print string"
       syscall                          # print prompt
        li $v0, 5                       # load syscode for "read integer"
       syscall                          # read next number (into $v0)
        move $t0, $v0                   # set Max to that value
next:   la $a0, Prompt                  # load prompt address
       li $v0, 4                        # load syscode for "print string"
       syscall                          # print prompt
       li $v0, 5                        # load syscode for "read integer"
       syscall                          # read next number (into $v0)
       ble $v0, $t0, cont               # check if < or  = Max
       move $t0, $v0                    # if not, update Max
cont:   addi $t1, $t1, 1                # increment counter
        ble $t1, $t2, next              # if < limit, do again
       la $a0, OutMssg                  # load output-message address
       li $v0, 4                        # load syscode for "print string"
        syscall                         # print output message
        move $a0, $t0                   # prepare argument for print
```

```
li $v0, 1                           # load syscode for "print integer"
 syscall                            # will print integer in $a0
li $v0, 10                          # load code for "stop
syscall                              # end execution
```