```
.data
   prompt: .asciiz "Please enter an integer:\n"
   message: .asciiz " is a triangular number."
   message1: .asciiz " is not a triangular number: "
   message2: .asciiz " , "
.text
   li $v0, 4  # getting a string load immediate
   la $a0, prompt # load address
   syscall

   # get user input
   li $v0, 5  # getting an int
   syscall

   #store result in $t0
   move $t0, $v0  # t0 is input

   #for loop
   main:
      addi $t1,$zero,1   #t1 is i, i=1

      beq  $t0,1,L1

      while:
         addi $t2,$t1,1 # t2=(1+t1)
         mult $t2,$t1   # t2*t1
         mflo $t3   # t3=t2*t1
         div  $t4,$t3,2 # t4=t3/2; t4 is the sum

         beq  $t4,$t0,Equal # t0=input; jump to Equal if t4==t0
         bgt  $t4,$t0,NotEqual  # jump to NotEqual if t4>t0

         addi $t1,$t1,1 # t1++
         j while       # jump back to while

      Equal:
         li $v0, 1  # print an int
         move $a0,$t4
         syscall

         li $v0, 4  # display as a string
         la $a0, message    # load address
         syscall

         j Exit

      NotEqual:
         sub $t5,$t4,$t1    # t5(the number less than input)=t4(the number bigger than
input)-t1(i)
         li $v0, 1  # print an int
         move $a0,$t0   # print t0(input)
         syscall

         li $v0, 4  # display as a string
         la $a0, message1   # load address
         syscall

         li $v0, 1  # print an int
         move $a0,$t5   # print smaller triangular number
         syscall
```

```
   li $v0, 4  # display as a string
   la $a0, message2   # load address
   syscall

   li $v0, 1  # print an int
   move $a0,$t4   # print bigger triangular number
   syscall

   j Exit

L1:
   li $v0, 1  # print an int
   move $a0,$t1   # print t1=1
   syscall

   li $v0, 4  # display as a string
   la $a0, message    # load address
   syscall

   j Exit

Exit:
   li $v0,10  # end program
   syscall
```