Task:

Write down a simple C program to find out the prime numbers between 1-10 and give the final count of prime numbers. Convert the C code into assembly and machine code and stimulate it on Venus.

C Code:

```c
int main() {
    // Write C code here
int p = 0;
int count = 0;
int z = 0;

    for(int i=2; i<=10; i++)
    {
        for(int j=2; j<=10; j++)
        {
            z = i % j;
            if(z==0)
            {
                p = p + 1;
            }
            else
            {
                p = p;
            }
        }
        if(p == 1)
        {
            count = count + 1;
            p = 0;
        }
        else
        {
            count = count;
            p = 0;
        }
    }
}
```

Assembly Code:

```
main:
    addi s0, x0, 0 # value of p
    addi s1, x0, 0 # value of count
    addi s2, x0, 0 # value of modulus variable
    addi t0, x0, 2 # value of i
    addi t1, x0, 2 # value of j
    addi t2, x0, 11 # value of condition for loop
    addi t3, x0, 1 # value to check for counting prime numbers
    j Loop_2

Loop_2:
    rem s2, t0, t1
    beq s2, x0, plus
    addi s0, s0, 0
    addi t1, t1, 1 # increment j loop
    blt t1, t2, Loop_2
    j Loop_1

plus:
    addi s0, s0, 1
    addi t1, t1, 1 # increment j loop
    j Loop_2

Loop_1:
    beq s0, t3, count
    addi s1, s1, 0 # count = count
    addi s0, x0, 0 # p = 0
    addi t0, t0, 1 # increment i loop
    addi t1, x0, 2 # initiaize j loop again
    blt t0, t2, Loop_2
    j end

count:
    addi s1, s1, 1 # count = count + 1
    addi s0, x0, 0 # p = 0
```

```
count:
    addi s1, s1, 1 # count = count + 1
    addi s0, x0, 0 # p = 0
    addi t0, t0, 1 # increment i loop
    addi t1, x0, 2 # initiaize j loop again
    blt t0, t2, Loop_2
    j Loop_2

end:
```