

```

# Title: Project 1 Part II                               Filename: Project 1 Part II.s
# Author: Dan Cassidy                                   Date: 2015-03-02
# Description: This program outputs the first 100 prime numbers.
# Input: Nothing
# Output: The first 100 prime numbers.
# Variables:
#   main: $s0 = numPrimes, $s1 = potentialPrime
#   test_prime: $a0 = n, $t1 = halfN, $t2 = i
##### Data segment #####
.data

##### Code segment #####
.text
.globl main
main:                                     #main program entry
    addi    $s7, $zero, 2                #load 2 because it's used a lot

    addi    $s0, $zero, 100              #set the number of primes to find (numPrimes)

    #2 is the only even prime number, so output that separately,
    #then only odds have to be checked for primeness
    addi    $s0, $s0, -1                 #decrement numPrimes because 2 is first prime
    li      $v0, 1                       #prepare to output 2
    addi    $a0, $s7, 0                  #set output to 2
    syscall                                #output 2
    li      $v0, 11                      #prepare to output a space
    addi    $a0, $zero, 32               #set output to a space
    syscall                                #output a space

    addi    $s1, $zero, 1                #initialize potentialPrime to 1
loop_m:    addi    $s1, $s1, 2            #increment potentialPrime by 2
    addi    $a0, $s1, 0                  #load argument for test_prime
    jal     test_prime                   #call test_prime to test potentialPrime
    beq     $v0, $zero, loop_m           #if (test_prime returns 0), jump to loop_m
    addi    $s0, $s0, -1                 #otherwise, decrement numPrimes (one less to find)
    #the following two statements aren't needed due to the way
    #values line up; they are kept in for reference only
    #li      $v0, 1                       #prepare to output potentialPrime
    #addi    $a0, $s1, 0                  #set output to potentialPrime
    #syscall                                #output potentialPrime
    #li      $v0, 11                      #prepare to output a space
    #addi    $a0, $zero, 32               #set output to a space
    #syscall                                #output a space
    bne     $s0, $zero, loop_m           #if (numPrimes != 0), jump to loop_m

exit_m:    li      $v0, 10                #prepare to exit program
    syscall                                #exit program

# Function: test_prime
# Description: Tests a number and determines whether it is prime.
# Input:
#   $a0, holds the number to be tested, must be odd and >= 3

```

```

# $v0, holds 1 if the number is a prime and 0 if not
#####
test_prime:                                     #test_prime function entry
    div      $a0, $s7                          #divide n by 2
    mflo     $t1                              #get n / 2

    addi     $t2, $zero, 3                    #set i to 3
    slt     $t0, $t1, $t2                    #set if (halfN < i)
    bne     $t0, $zero, exit_t               #if (halfN < i)[i <= halfN], jump to exit_t
loop_t:    div      $a0, $t2                  #divide n / i
    mfhi     $t0                              #get n % i
    bne     $t0, $zero, skip_t               #if (n % i != 0), jump to skip_t
    addi     $v0, $zero, 0                   #set return value to false
    jr      $ra                              #return to main
skip_t:    addi     $t2, $t2, 2               #increment i by 2
    slt     $t0, $t1, $t2                    #set if (halfN < i)
    beq     $t0, $zero, loop_t               #if (i <= halfN), jump to loop_t

exit_t:    addi     $v0, $zero, 1             #set return value to true
    jr      $ra                              #return to main

```