

- Programming Assignment #5 -- (*Calculate the nth Prime Number*)
- Due: see canvas

Name your program: p5.c

This week we are going to simplify last week's program with the use of a Function. The code below is the code shown in class last week. This week we need to replace the code in the black rectangle with a function called, `check_if_prime()`;

```
#include <stdio.h>

#define TRUE 1
#define FALSE 0

//function prototype goes here

int main (void) {

    unsigned int    nth_prime, number_of_primes;
    unsigned int    test_int, divisor;
    _Bool           isPrime;

    printf("Enter the nth_prime to find:");
    scanf("%u", &nth_prime);

    number_of_primes = 0;
    test_int = 1;

    do
    {
        test_int++;

        isPrime=TRUE;    //assume test_int is prime until proven not to be
        for(divisor = 2; divisor < test_int; divisor++)
        {
            if(test_int % divisor == 0)
            {
                isPrime = FALSE;
                break;
            }
        }

        if(isPrime == TRUE)
            number_of_primes++;

    }
    while( number_of_primes < nth_prime );

    printf("The %u Prime Number = %u\n", nth_prime, test_int);
    return 0;
}
```

The function definition should look like:

```
_Bool check_if_prime( unsigned int test_number)
{
    // code to determine if a number is prime, goes here
}
```

This function takes an (unsigned int) as an argument, and returns 0 (i.e. false) if the integer is not prime, and returns 1 (for true) if the integer is prime.

Once you've completed your program, fill in the table below.

600 th Prime Number is:	6,000 th Prime Number is:	60,000 th Prime Number is:

Use the following command to submit your p5.c code
cp p5.c /home/faculty/skoss/cse121/your_UID
or
./submit p5.c

Also, fill in the table above, and hand it in the following (Monday).
