

- 6.5 Prime numbers are integers greater than 1 with no positive integral divisors except for themselves and 1. The following program consists of a function *int IsPrime (int Num)* to determine if the integer Num is a prime number or not. Try to walk through the program and study the function.

```
/* This program shows the use of user-defined functions to determine
   if a number is a prime number or not.*/

#include <stdio.h>

int IsPrime (int Num);

int main (void)
{
    int Num;
    char Choice;

    printf("\nStart program.\n");
    do
    {
        printf("\nPlease enter an integer : ");
        scanf("%d", &Num);
        if (IsPrime(Num))
            printf("\nthe number %d is prime", Num);
        else printf("\nthe number %d is not prime", Num);
        printf("\n");
        printf("\nDo you want to try again (y/n)? ");
        scanf(" %c", &Choice);
    } while ((Choice == 'y') || (Choice == 'Y'));

    printf("\nEnd program.");
    printf("\n");
    return 0;
}

int IsPrime (int Num)
{
    int SoFarPrime;
    int i;

    for (SoFarPrime = 1, i=2; i <= Num-1; i++)
        if ((Num % i) == 0)
            SoFarPrime = 0;
    return SoFarPrime;
}
```

Task : Making use of the function *int IsPrime (int Num)*, try to write a function with an integer as input parameter and display all the prime factors of the integer. Write a program to test your function.

Sampling running :

Start program.

Please enter an integer : 3<CR>

The number 3 is prime.

Do you want to try again (y/n)? y<CR>

Please enter an integer : 4<CR>

The prime factor(s) of 4 is/are
2

Do you want to try again (y/n)? y<CR>

Please enter an integer : 70<CR>

The prime factor(s) of 70 is/are
2 5 7

Do you want to try again (y/n)? y<CR>

Please enter an integer : 1517<CR>

The prime factor(s) of 1517 is/are
37 41

Do you want to try again (y/n)? n<CR>

End program.