```
Select operation
 1. Calculate sum/multiplication between two numbers
 2. Calculate prime numbers
 3. Show number sequence in file
 4. Sort number sequence in file
 ------------------------
```

In this function, the program respectively request a two Integer (N1, N2), an operation flag, and finally a flag to decide if it is going to work on odd / even numbers. The program will support only 2 operations, addition or multiplication. According to the selection of the flags, the program will calculate the sum/multiplication of the odd/even numbers between the range of [N1,N2]. The addition and the multiplication operations should be done by using two different functions. The operation selection should be determined by using operation flag with switch-case, after that, the integer and the other flag should be used to call the related function. If user enters invalid value for the flags, the program should print an error message. After each calculation process is completed, the result should be written to the file named "results.txt".

Function prototypes are :

int sum (int n1, int n2, int flag)

int multi (int n1, int n2, int flag)

Examples:

```
Select operation
Please enter '0' for sum, '1' for multiplication.
0
Please enter '0' to work on even numbers, '1' to work on odd numbers.
0
Please enter two different number:
Number 1: 1
Number 2: 11
Result
2 + 4 + 6 + 8 + 10 = 30
The result is written to the results.txt file.
```

```
 1
Select operation
Please enter '0' for sum, '1' for multiplication.
1
Please enter '0' to work on even numbers, '1' to work on odd numbers.
1
Please enter two different number:
Number 1: 1
Number 2: 11
Result
 3 * 5 * 7 * 9 = 945
The result is written to the results.txt file.
```

Write a function which takes an integer N from the user and checks the every integer from 2 to that number if they are prime or not. The primality testing is made by a function with the following information:

A is prime if A is not dividible by any integer X where X is;

$$1 < X \leq \sqrt{A}$$

This operation should be done in a function, the function should return a flag if the integer is prime, or should return the least divisor of that integer if it is not a prime. In the main function, you should use a loop to check every A between $1 < A < N$ obtain a result by using the function and finally print it.

Only 'for' loops should be used. You are allowed to use sqrt() function from math library to calculate the square root.
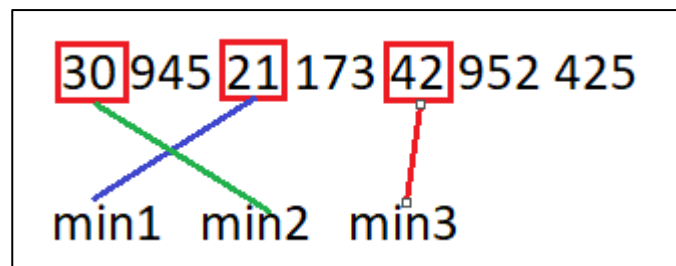
Function prototype is :

int isprime (int a)

Expected Output

```
Please enter an integer: 13
2 is a prime number.
3 is a prime number.
4 is not a prime number, it is dividible by 2.
5 is a prime number.
6 is not a prime number, it is dividible by 2.
7 is a prime number.
8 is not a prime number, it is dividible by 2.
9 is not a prime number, it is dividible by 3.
10 is not a prime number, it is dividible by 2.
11 is a prime number.
12 is not a prime number, it is dividible by 2.
```

In file operations, you must create a function named write_file() for writing to the file. This function should take the value to be written to the file as a parameter. The 3rd option on the menu should call the print_file() function and print the numbers in the "results.txt" to the terminal. The 4th option in the menu should call the function named sort_file(). In this function, a simple sorting operation should be performed in ascending order in the "results.txt" file. In this sorting process, you need to sort the mixed numbers in the results.txt file from smallest to largest. You should select 3 numbers in each step. You should determine the smallest three numbers in sequence and keep them on a temporary file. In other words, you should assign the smallest numbers to the min1, min2, min3 variables and keep them in order from smallest to largest in a temporary file. A sample image of this is below. In each sorting round, the smallest 3 numbers should be determined and transferred to the temporary file. You are not allowed to use arrays. You can create the necessary functions by adding your comment lines. When you complete all the operations, you should print all the numbers in order from smallest to largest on "results.txt". Remember, you need to sort the numbers 3 by 3, not one by one.

Function prototypes are:

void write_file (int number)

void sort_file ()

void print_file ()

Expected Outputs :