

CSE108 – Computer Programming Laboratory

Lab #4

Date: Monday March 04, 2019

Handin: A student with number 20180000001 should hand in three separate files named 20180000001_par1.c, 20180000001_par2.c, 20180000001_par3.c and 20180000001_par4.c for this lab.

Part 1. [30pts] Write a function that takes one integer argument to calculate all prime numbers which are smaller than the given number. The function should print founded prime numbers and number of total prime numbers that are under the given number. An example to the output of the function has given below. The function header is:

```
void find_prime_numbers(int number)
```

Output:

```
Please enter a number to find prime numbers that are smaller than the number:
100
2 3 5 7 11 13 17 19 23 29 31 37 41 43 47 53 59 61 67 71 73 79 83 89 97
There are 25 prime numbers which are smaller than 100
```

Part 2. [30pts] Let's assume that there is a grid that is shown below. An ant starts for a trip from (0, 0) point to reach up ant nest. Please note that, the nest appears randomly on each trip. The ant can move vertical, horizontal and diagonal between points of the grid. The distances between two points of the grid in horizontally, vertically and diagonal are 3,4 and 5 respectively. Write a program that simulates the trip that is mentioned above. The program takes movements of the ant from user until it reach up the nest. The movements must be pre-defined as; Left:4, Right:6, Up:8, Bottom: 2, Left and Bottom: 1, Right and Bottom: 3, Left and Top:7, Right and Top:9. Call the function which is stated on Part 3 after the each move has done. After the ant reach up its nest, the program prints the total distance of the trip. Please note that, the ant can't violate borders of the grid, if the user wants to make a move which is not allowed, print a message to the user and ignore the forbidden move.

```

X      .      .      .      .      .      .      .      .      .
.      .      .      .      .      .      .      .      .
.      .      .      .      .      .      .      .      .
.      .      .      .      .      .      .      .      .
.      .      .      .      .      .      .      .      .
.      .      .      .      .      .      .      .      .
.      .      .      .      .      .      .      .      .
.      .      .      .      .      .      .      .      .
.      .      .      .      .      .      .      .      .
.      .      .      .      .      .      .      .      .

```

* X represents the ant and 0 represents the nest.

Part 3. [10pts] Write a function that prints the grid in the format which is shown on the part 2 according to current position of the ant and the nest. The "X" stands for the ant and the "O" represents the nest. The function header is given below.

```
void print_the_grid(int x_pos_ant, int y_pos_ant, int x_pos_nest, int  
                    y_pos_nest)
```

Part 4. [30pts] Write a function that brings the ant to the nest by using minimum path between the ant and the nest and returns distance of the minimum path. The minimum path can be found by following two rules:

- Make a diagonal move towards the nest until the ant and the nest are not on the same vertical or horizontal line.
- If the ant and the nest are on the same vertical line, do vertical move towards the nest until the ant reaching up to the nest.
- If the ant and the nest are on the same horizontal line, do horizontal move towards the nest until the ant reaching up to the nest.

The function should print the grid after each move has done by using the function that is stated on Part 3. The function header is given below.

```
int find_min_path(int x_pos_ant, int y_pos_ant, int x_pos_nest, int  
                  y_pos_nest)
```