Assignment 3 – User-Defined Functions/Arrays

Deadline: **Monday Oct. 26 by 11:55pm**
Type: **Group Assignment**
Weight: **This assignment is worth 5% of your final grade**
Submission: **Must be on Moodle**

Notes:
- Please do not submit exe files
- All submissions must be done through Moodle

Marking Scheme:
- Program correctness (80%)
- Program clarity (output format, comments, completeness, readability) (20%)

Exercises:

1. **(25 marks)** Write a **menu-driven** program that prompts the user to enter an integer between 10 and 999 and allows the user to perform one or more of the following operations:
   - Display the digits that compose the number (e.g., if the user enters 215 then the program displays 2 – 1 – 5)
   - Reverse the digits of the number (e.g., 215 becomes 512)
   - Sum the digits of the number (e.g., for 215, the result is 8 = 2+1+5
   - Returns true if at least one of the digits is a prime number

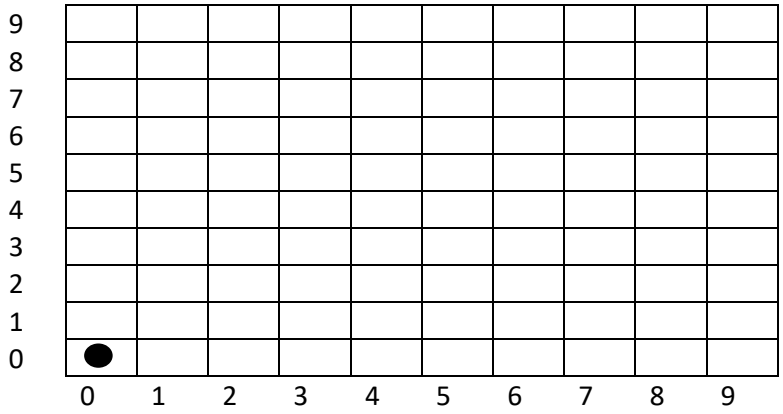
Your program should implement each of these operations in a separate function. Use the appropriate parameters to pass information in and out of a function.

2. **(25 marks)** Write a C++ program that takes a text as input and returns the following information:
   - The total number of words
   - The number of personal pronouns: I, you, he, she, it, we, they, me, him, her, us, and them
   - The number of punctuation points: comma, period, exclamation mark, question mark, colon, semicolon
   - The number of words that start with a vowel
   - The number of words that start with a consonant

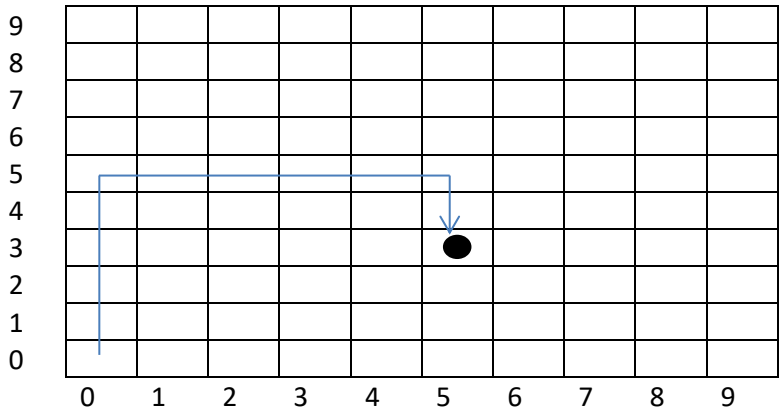
Assume the text can have a maximum of 100 words. Use functions and arrays as needed.

3. **(50 marks)** We want to create a program that controls the movements of an object in a 10x10 grid. Initially, the object is at position (0, 0). The controller supports the following commands:
   - up(n): the object moves n cells towards the top of the grid
   - down(n): the object moves n cells towards the bottom
   - turn right (n): the object turns right and moves n cells
   - turn left (n): the object turns right and moves n cells
   - reboot: the object comes back to cell (0, 0)

Consider the following grid:



The new grid after executing the sequence: up(5), turn_right(5), down(2) will look like this:



The program takes commands from a user as shown in the following example:

Controller Menu:

1. Up
2. Down
3. Turn Right
4. Turn Left
5. Reboot
6. Show Grid
7. Exit

Command "6. Show Grid" displays the grid. Command "7. Exit" allows exiting the program.

A. Implement the controller's functionalities. Add necessarily checks to ensure that the object does not go out of the grid. Your program should be structured using functions. For example, each command should be implemented in a separate function. (25 marks)

B. We want to add the following commands (25 marks):
   - cancel (n): The program cancel the last n operations and return the object to the initial position
   - replay (n): The program replays the last n operations that have been cancelled.
   - position: The problem displays the position of the object

C. **(Bonus)** Add a second object to the grid. Repeat questions A and B. The two objects should never collide, meaning, they should never end up on the same cell (25 marks).