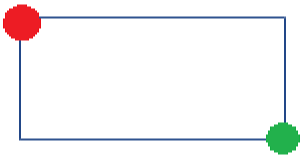
MIDDLE EAST TECHNICAL UNIVERSITY, NORTHERN CYPRUS CAMPUS
CNG140 C Programming – Worksheet 24

Purpose:
In this work sheet, you will get hands-on experience with structures and files.

Tasks:

a) Assume that you have a structure to represent a rectangle. For this structure you need to keep track of the coordinates of the two corners of a given rectangle. You need to keep track of the upper left x, y coordinates (shown as red in the below figure) and lower right x, y coordinates (shown as green in the below figure). Write functions that perform the following operations on a rectangle structure r passed as an argument:



i) Read the upper left x, y coordinates and lower right x, y coordinates of the rectangle (r) from the file called as "rectangle.txt".
Assume "rectangle.txt" contains data as follows:
3 8
9 2

ii) Compute the area of r;

iii) Compute the center of r (represent as x and y coordinates). If either x or y coordinates of the center isn't an integer, store its truncated value in the point structure. Assume that x=1.99, then after truncation x=1.You need to use pass by reference for this operation.

A sample run can be as follows:

```
"rectangle.txt" is successfully read!
upper left x coordinate is 3 and y coordinate is 8
lower right x coordinate is 9 and y coordinate is 2
area of the rectangle is 36
center x coordinate of the rectangle is 6
center y coordinate of the rectangle is 5
```

b) Write a program that includes the following operations. In your program define a global data structure as follows:

```
struct Person {
      char lastname [15];
      char firstname [15];
      char age[4]; };
```

i) Create an array of 100 Person in your main function;

ii) Initialize this array with the following default values: lastname="unassigned", firstname="unassigned" and age ="0";

iii) Create a function called add() – This function takes the array of Person structure defined and initialized in part (ii) and asks the user to enter the details of a Person and adds this person to the array. If the array is already full, this function should say that you cannot add a new Person. Please note that you can store at most 100 Person.

iv) Create a function called search() – This function will take the array created in part (ii) and the lastname of the person from the user and searches that user in your array. If it is found, it displays the values to the user, if not it says "No user found! "

v) Create a function called delete() – This function will take the array created in part (ii) and the lastname of the person from the user, and searches that user in the array. If the user exists, it deletes from the array.

vi) Create a function called makePersistent() – This function takes the array and writes the content to an external file called "external.txt".

A sample run can be as follows: Your program should initialize the file and then provide a menu for the functions listed above (ii, iii, iv,v) so that the user can choose which one to process.

```
1) Add person
2) Search person
3) Delete person
4) Store the content
5) Exit
What would you like to do? 1
Enter the name of the person: John
Enter the lastname of the person: Black
Enter the age of the person: 32

1) Add person
2) Search person
3) Delete person
4) Store the content
5) Exit
What would you like to do? 2
Enter the lastname of the person: Black
Name and Surname: John Black Age: 32

1) Add person
2) Search person
3) Delete person
4) Store the content
5) Exit
What would you like to do? 3
Enter the lastname of the person: Black
Name and Surname: John Black Age: 32 is deleted!

1) Add person
2) Search person
3) Delete person
4) Store the content
5) Exit
What would you like to do? 2
Enter the lastname of the person: Black
No user found!

1) Add person
2) Search person
3) Delete person
4) Store the content
5) Exit
What would you like to do? 4
The content is stored to an external file called "external.txt".
```

```
1) Add person
2) Search person
3) Delete person
4) Store the content
5) Exit
What would you like to do? 5
```

c) Write a data structure to represent a movie. In this structure, you need to keep track of the name of the movie and the ranking (integer number that can be between 0-10). Define an array of 10 movie structures that can be used to store information about the movies. Write functions that perform the following operations:

i) **addMovie()** – This function will take the array of movie structure created, and also the number of movies stored in the array and add a new movie to the array. Please note that initially, the size of the array is 10. If the number of movies is already 10, then the main function should reallocate the size of the array such that a new movie can be added to this array.

ii) **findBest()** – This function takes the array of movies structure created, the number of movies stored and finds the movie with the highest ranking in the list.

iii) **storeMovie()** – This function takes the array of movie structure created and the size of the list, and stores them to a file called “movie.txt”.

A sample run would be as follows:

```
1) Add movie
2) Find best movie
3) Store movies
What would you like to do? 1
Enter the name of the movie: Yol
Enter ranking: 10
Yol is added!

1) Add movie
2) Find best movie
3) Store movies
What would you like to do? 1
Enter the name of the movie: Vizontele
Enter ranking: 8
Vizontele is added!

1) Add movie
2) Find best movie
3) Store movies
What would you like to do? 2
Yol is the best movie!

1) Add movie
2) Find best movie
3) Store movies
What would you like to do? 3
Movie.txt is successfully created!
```

d) Write a data structure to represent a circle. In this structure, you need to keep track of the radius, the x-coordinate and the y-coordinate of a circle. Define an array of 10 circle structures that can be used to store information about the circles. Write functions that perform the following operations:

i) **addCircle()** – This function will take the array of circle structure created, and also the number of circles stored in the array and add a new circle to the array. The function will ask user to enter the x-y coordinates and also the radius of the circle. Please note that initially, the size of the array is 0 (array is empty). If the array is already full which means there are 10 circles already then you need to give the necessary warnings to the user.

ii) **searchPoints()** – This function takes the array of circle structure created, the number of circles stored, and inside the function gets a x-coordinate and y-coordinate of a point from the user, then finds if that entered point lies on any circle or not using the below formula. If points lie on any circle it will display circle information, otherwise it will display "This point does not lie on available circles!".

$$(\boldsymbol{x}_{\boldsymbol{circle}} - \boldsymbol{x}_{\boldsymbol{point}})^2 + (\boldsymbol{y}_{\boldsymbol{circle}} - \boldsymbol{y}_{\boldsymbol{point}})^2 == (\boldsymbol{r}_{\boldsymbol{circle}})^2$$

iii) **storeCircles()** – This function takes the array of circle structure created and the number of circles stored, and writes the circles in the array to a file called "circles.txt".

A sample run would be as follows:

```
1) Add circle
2) Search points
3) Store circles
4) Exit
What would you like to do? 1
Enter the x-coordinate of circle: 5
Enter the y-coordinate of circle: 5
Enter the radius of circle: 5

1) Add circle
2) Search points
3) Store circles
4) Exit
What would you like to do? 1
Enter the x-coordinate of circle: -5
Enter the y-coordinate of circle: 5
Enter the radius of circle: 5

1) Add circle
2) Search points
3) Store circles
4) Exit
What would you like to do? 2
Enter the x-coordinate of point: 0
Enter the y-coordinate of point: 0
This point does not lie on available circles!

1) Add circle
2) Search points
3) Store circles
4) Exit
What would you like to do? 2
Enter the x-coordinate of point: 0
Enter the y-coordinate of point: 5
```

```
This point lies on the following circles:
circle: x=5, y=5, r=5
circle: x=-5, y=5, r=5

1) Add circle
2) Search points
3) Store circles
4) Exit
What would you like to do? 2
Enter the x-coordinate of point: 5
Enter the y-coordinate of point: 0
This point lies on;
circle: x=5, y=5, r=5

1) Add circle
2) Search points
3) Store circles
4) Exit
What would you like to do? 3
circles.txt is successfully created!

…
…

1) Add circle
2) Search points
3) Store circles
4) Exit
What would you like to do? 1
Your array is full now and you cannot add a circle!


1) Add circle
2) Search points
3) Store circles
4) Exit
What would you like to do? 4
```

Recommended Reading: Chapter 12 (p. 577-597)
Recommended Exercises: Programming Exercises given in the following pages.