2021

Code Documentation

BATTLESHIP GAME DESIGNED IN C LANGUAGE.

JASMEET SINGH MATTA

Contents

Data Types Used

INT

Int is a data type in C.
It is of 2 or 4 bytes.
It can range from -32,768 to 32,767 or -2,147,483,648 to 2,147,483,647 (unsigned int)

CHAR

Char is a data type in C.
It is of 1 byte.
It can range from -128 to 127 or 0 to 255.

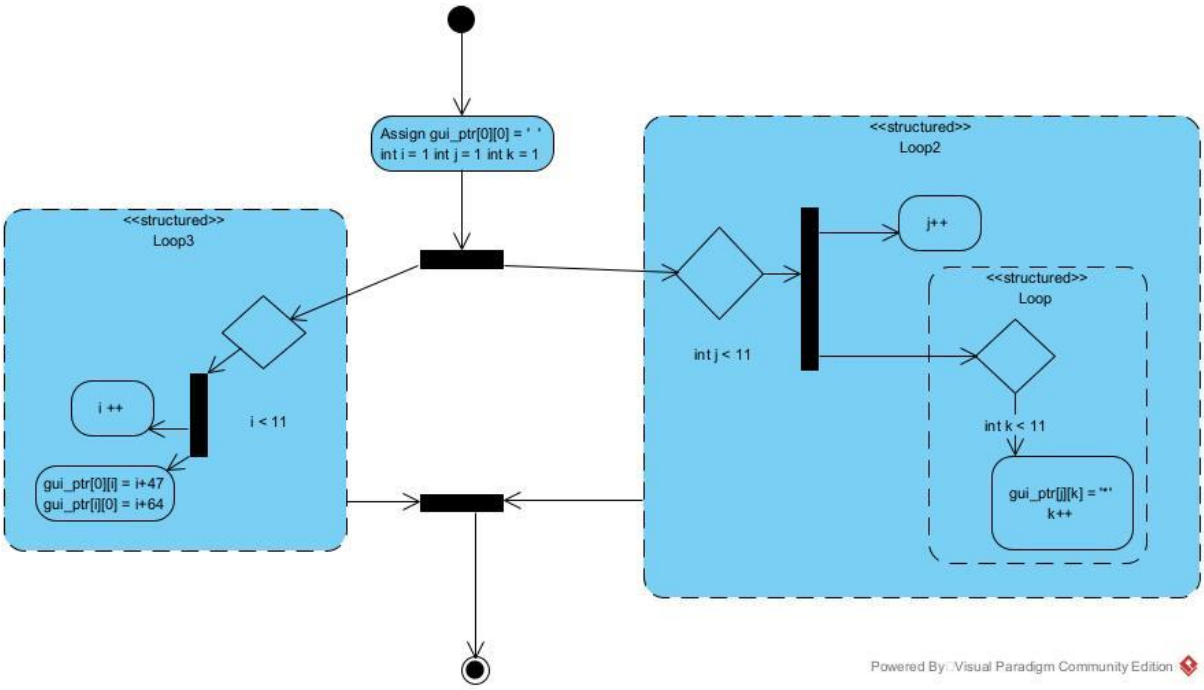
POINTERS

In the code you can see many variables define by either a single asterisk * or double asterisk **.
They are known as pointers or pointers to pointer, respectively.
You can think of them as array of single or multiple dimensions.
It easy to use as you can change value globally in code.

Functions

*void boardGui(char **gui_Ptr){…}*

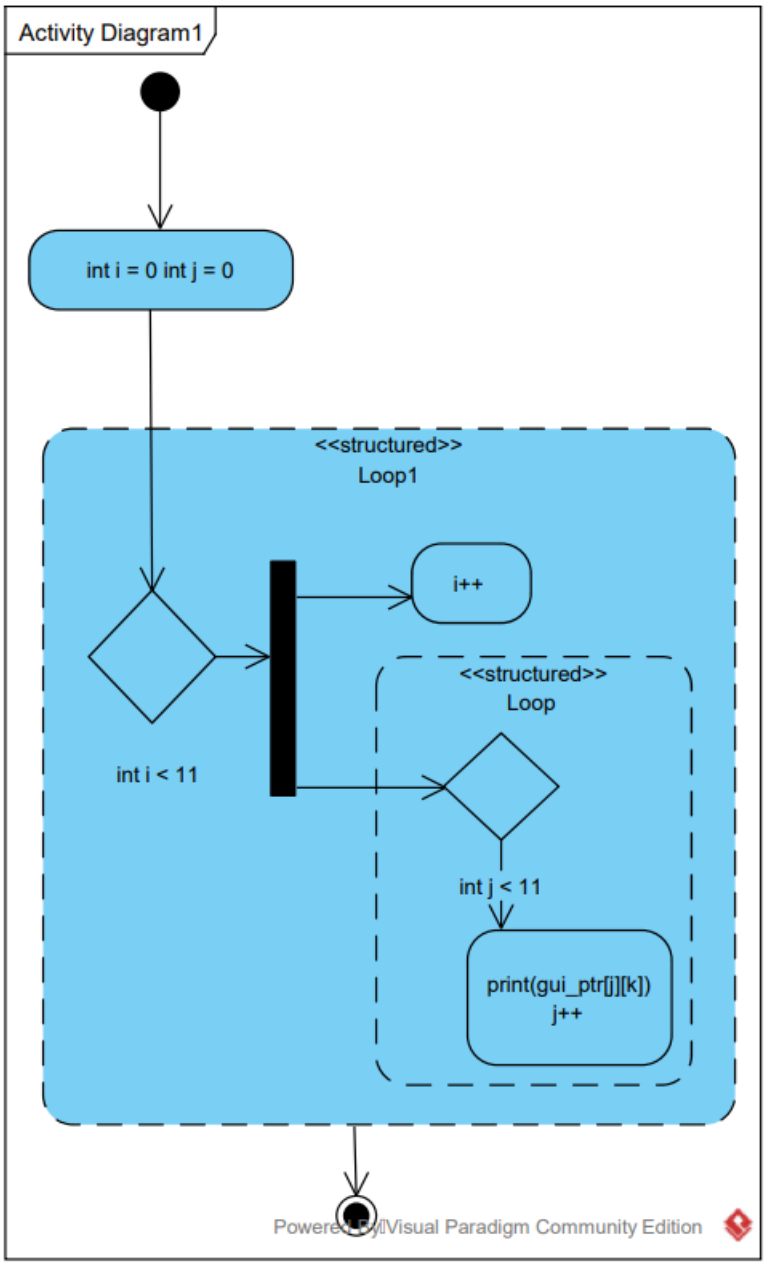
It is used to generate a board for our game.
It's a 11*11 board.
The space [0,val] and [val,0] are used by alphabets A-J and Numbers 0-9.
They will be used as reference to certain block in board.
Such as A,0 can be understood as Row A column 0.
It is used to place ship on board, visualise our ship placement and to shoot ship.



*void drawGui(char **gui_Ptr){…}*

It is used to Draw our board on Terminal.

It 11*11 board on terminal



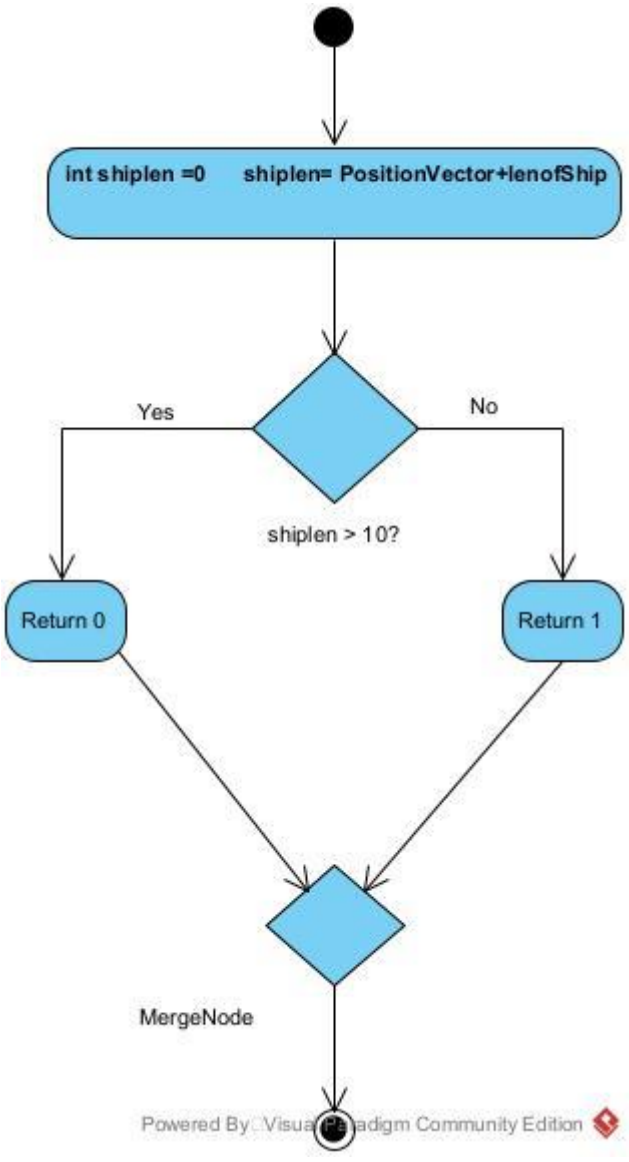
int checkCorners(int shipPointer,int s){…}

It is to check that if placing the ship at certain location and alignment will get it out of bounds.

Example if placing a Destroyer (A type which occupies 5 space of block) at A,0 and A,6.

At A,0 the function will return value of 1 as it can be placed.

At A,6 the function will return value of 0 as it can not be placed there as it is going beyond boundary. Which will give us core dump error.

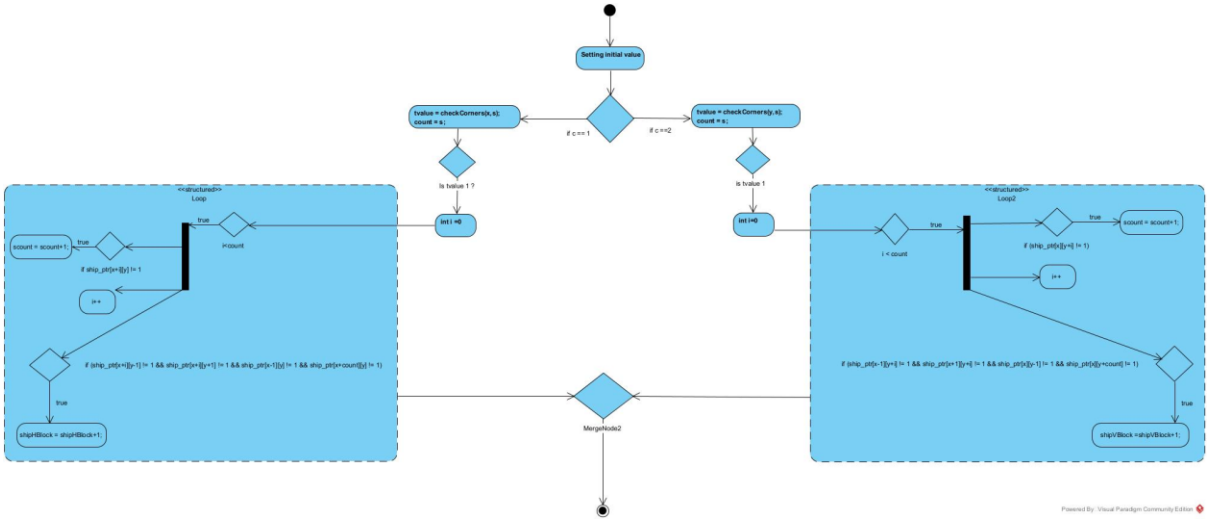


*int checkAllot(int **ship_ptr,int x,int y,int s,int c){…}*

It is to check that placing a ship at certain location is possible such as to see if there are other ship colliding and to see if there is a block gap between two ships.

If ship can be placed, it will return the value of 1.

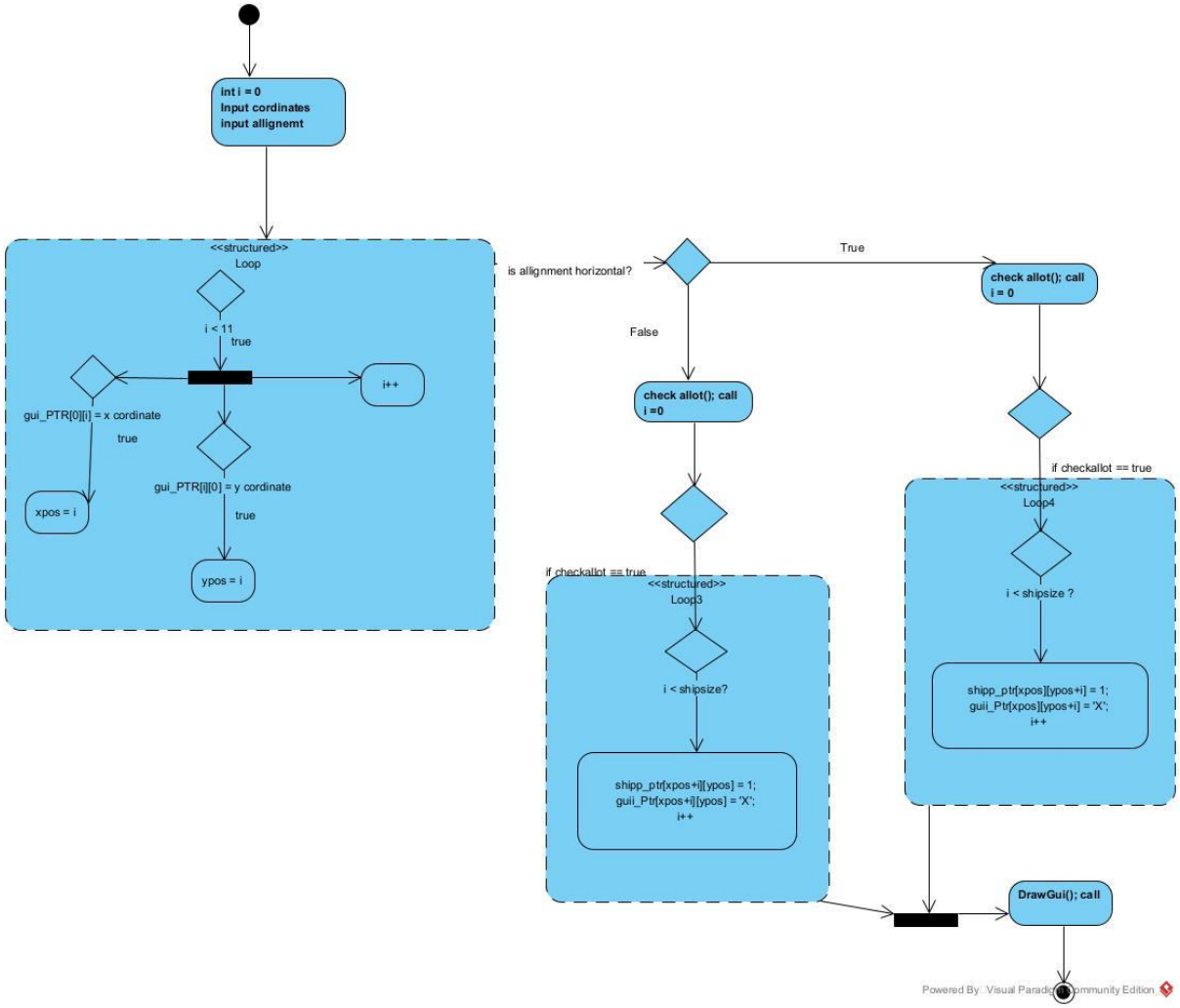
If ship cannot be placed due to any reason mentioned above it will return the value of 0.



*int placingShip(int **shipp_ptr,char **guii_Ptr,int *shipp_def ,int n){…}*

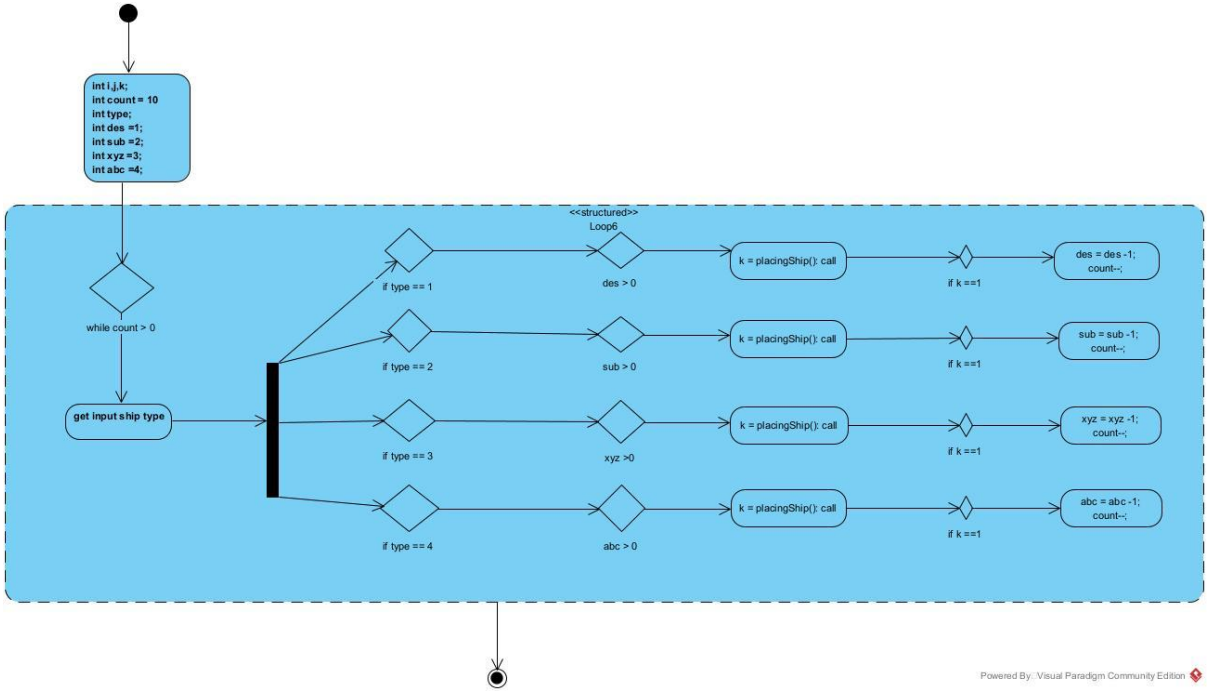
It is to take user input of coordinates and ship alignment.

After taking user input the function check for possibilities of ship placement and once it is confirmed that ship can be placed it changes the ship pointer value at X, Y to 1 and grid GUI to X for user reference.



*void shipPlacement(int **ship_Ptr,char **gui_ptr,int *ship_def){…}*

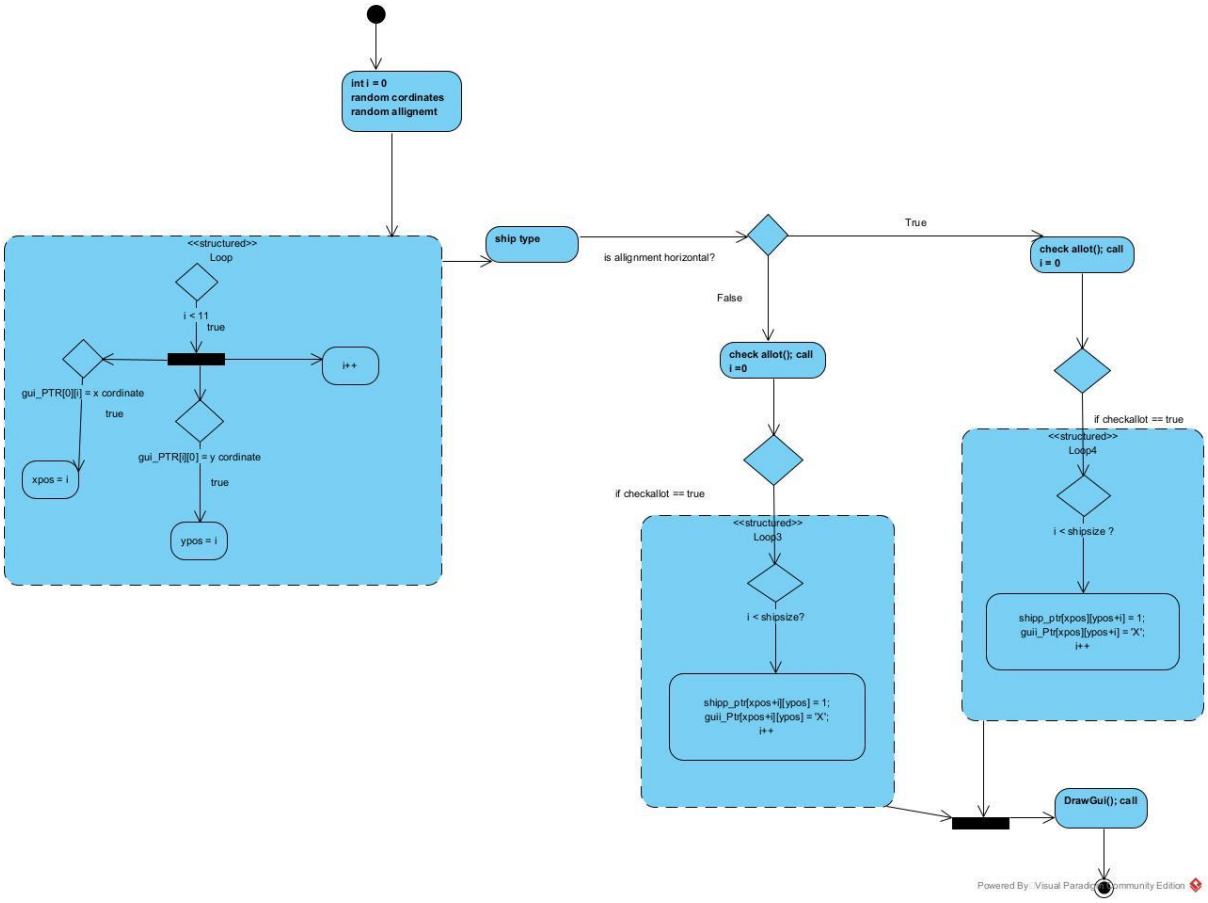
It is like a control unit of all function which are related to placing ship as this function defines the number of ship that can be placed and handle any error regarding placement such as if a ship can be placed 1 time only user cannot place that ship again.



*void shipPlacementComputer(int **shipCom_Ptr,int *ship_def,char **comGui_Ptr){…}*

This function is used to place ship randomly on board by computer here we create a random seed by calling a inbuilt srand() function and then use rand() function to generate random number.

Random number generated are converted into range from 0-9(10) for placing ship on board.



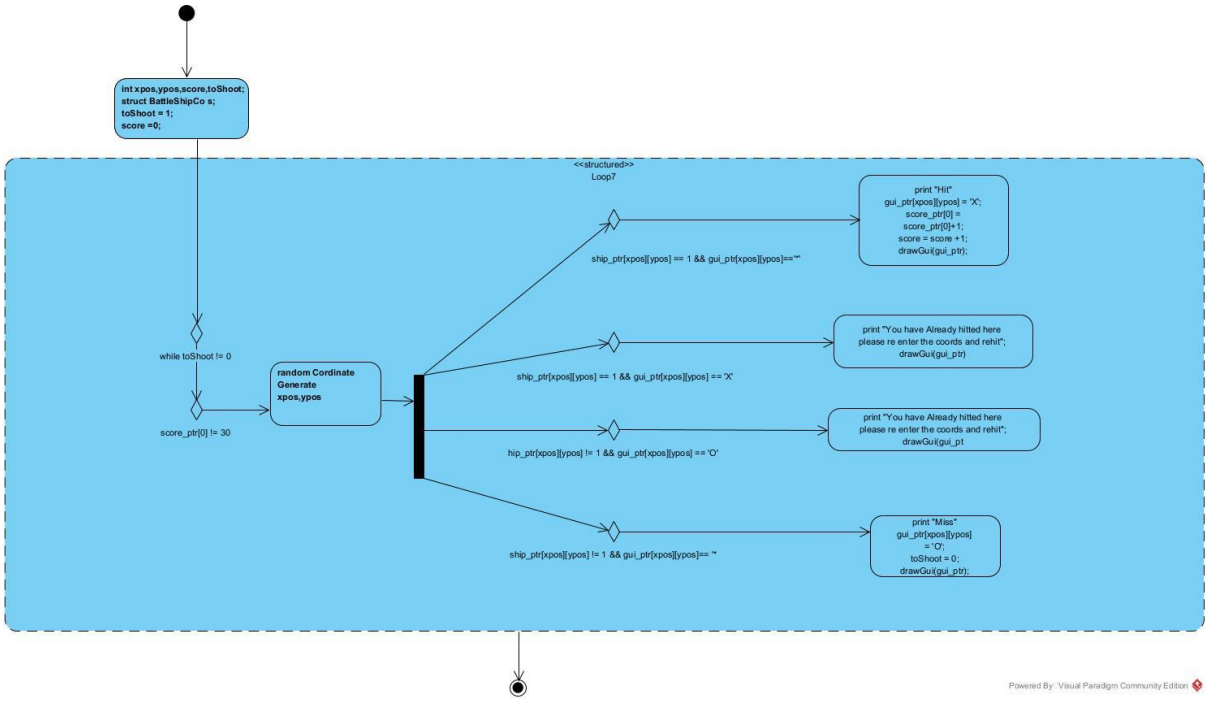
*int shootShip(char **gui_ptr,int **ship_ptr,int *score_ptr){…}*

This function takes user input for desired coordinate to shoot it take inputs till user enter valid coordinate such as X cord in Range from A-J and Y cord in Range from 0-9.
After Taking coordinate value this function checks if user has shoot at that position before and if not then tells user if he/she/other hit a ship or not.
If user hit a ship, he/she/other is allowed to shoot again till he/she/other misses.

*int comShootShip(char **gui_ptr,int **ship_ptr,int *score_ptr){…}*

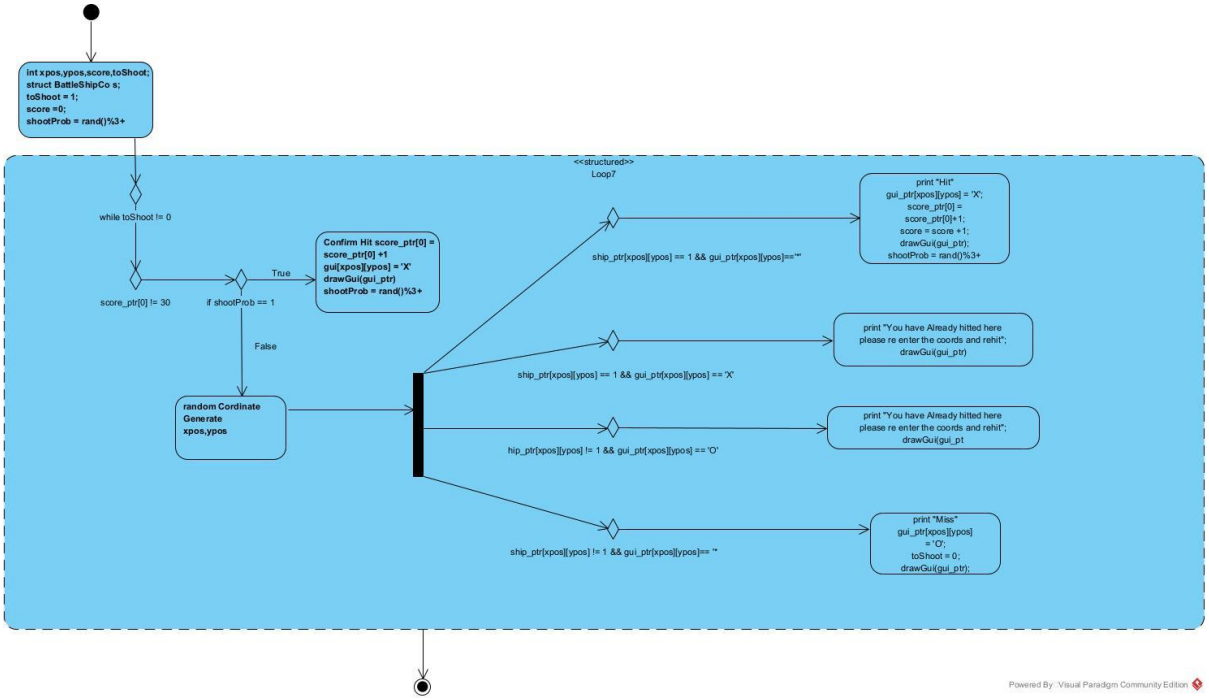
In this function computer generate a random coordinate and shoot at that coordinate here computer has Number of blocks occupied by ship/Number of ship available probability to shoot at ship. Therefore, this function is used as easy mode for user.

The only difference between com and player is in computer the coordinates are generated randomly.



*int comShootShip_Lvl2(char **gui_ptr,int **ship_ptr,int *score_ptr){…}*

In this function computer first generate a random number in the range of {1-3}.
The range can be changed through source and then recompiling the code.
If random number generated is 1 the probability to hit a ship is 100 percent.
If random number generated is other than 1 than probability to hit a ship is Number of block occupied by ship/Number of ship available.



int main(){…}

It is used to initialize all the necessary data type variable passed on to other function and is a main control unit of the code.
It ask user if he wants to play with computer or with other player and also explains rule/details about the game.

Code

Why such implementation?

a. As I was working alone, I decided to work with pointers to pointer as for new experience.
b. Wanted to have deep knowledge of pointers and experience handling them.
c. I dedicated a fix amount of time per week for writing code till deadline.

Tools and Technology

a. I used Ubuntu 18.04 OS for the code as I like to code on Linux based system.
b. I used Visual Studio Code for writing actual code.
c. GCC 10.3 was used to compile the code.

Structure

Implemented two grid of 11*11 one for storing ship data and other for printing playable board.
Keeping in mind of 11*11 Structure whole game was designed and build.
Used ASCII value for easy manipulation, integrity, and error handling.
ASCII character A-J were converted to number 1-10 in the implementation.
There are a few dummy implementation in the code for Footprint/Signature for the code.

GitHub: https://github.com/Jasmeet03/BattleShip/