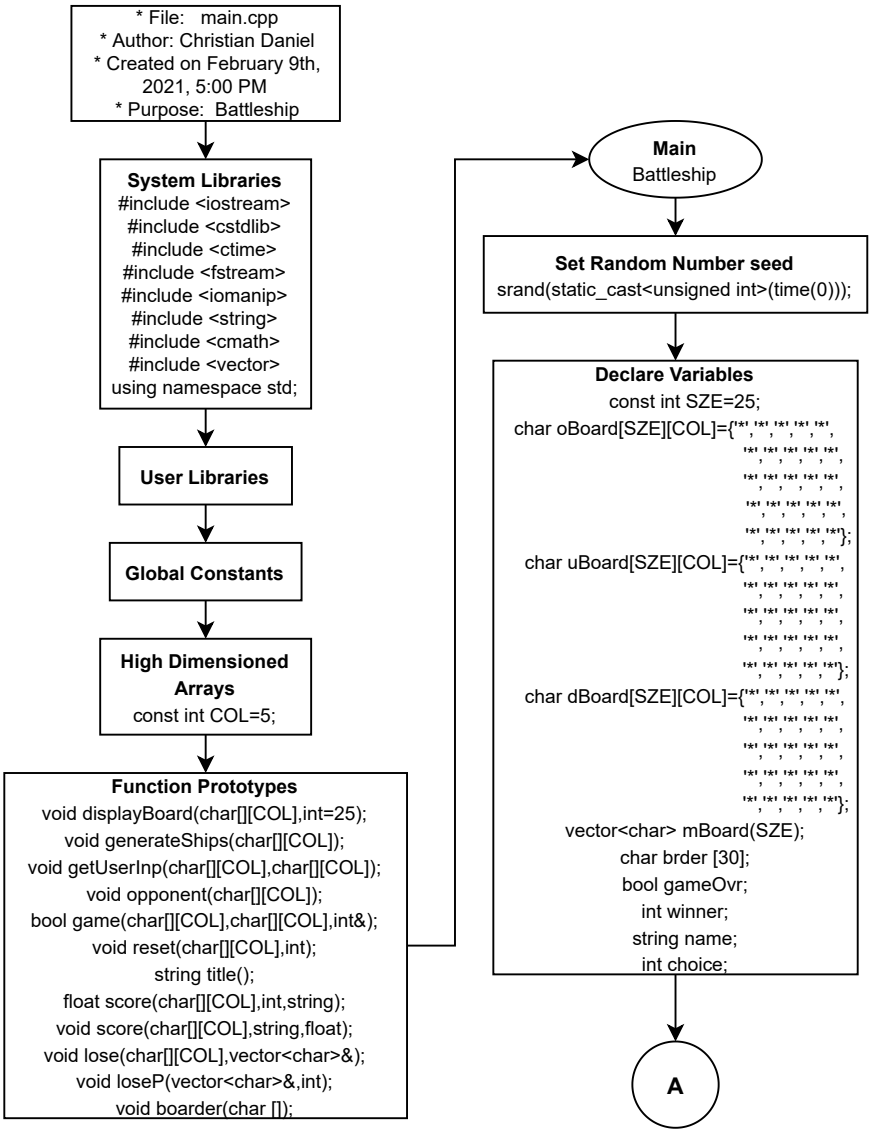
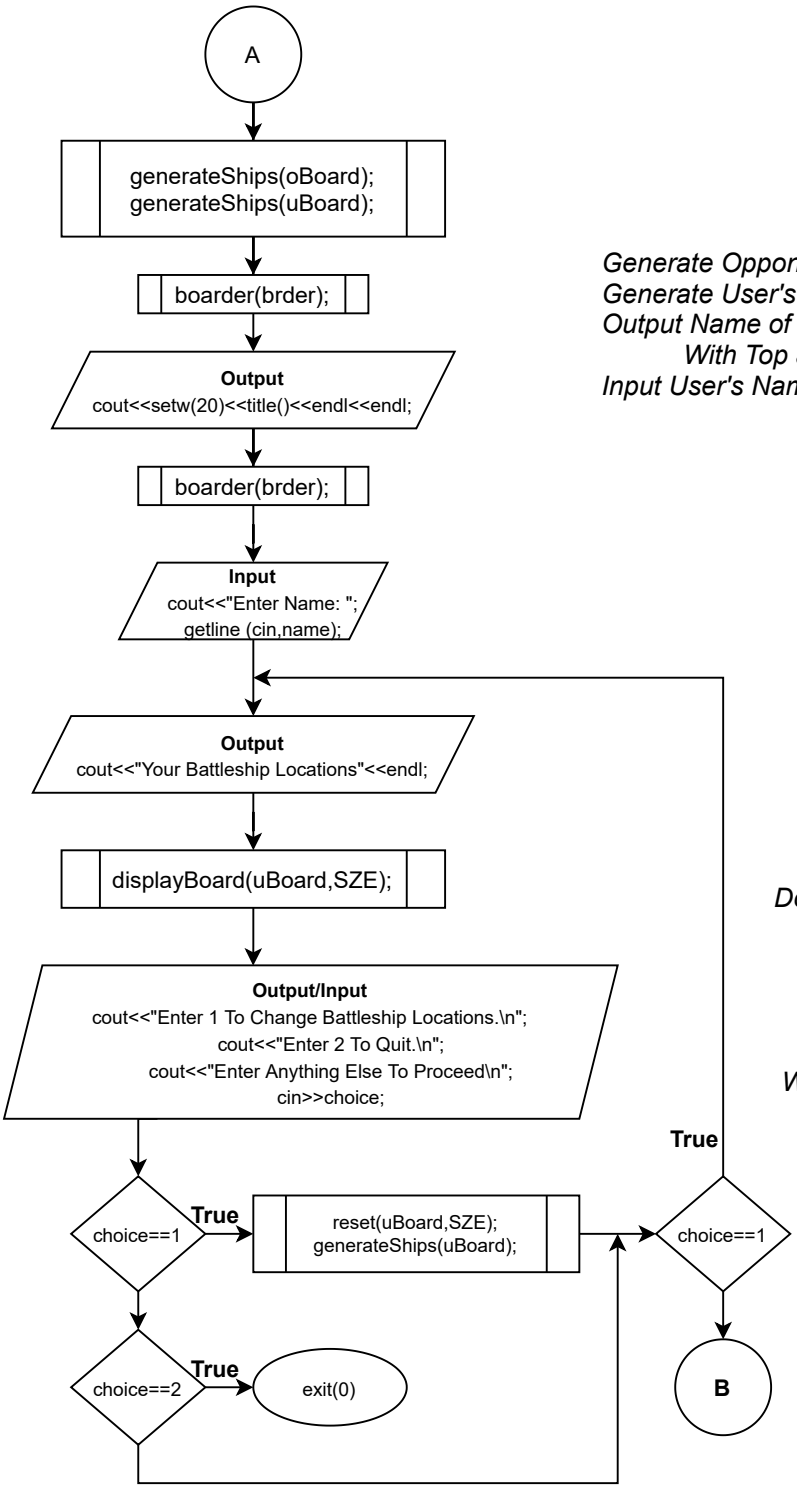
Battleship Flowchart



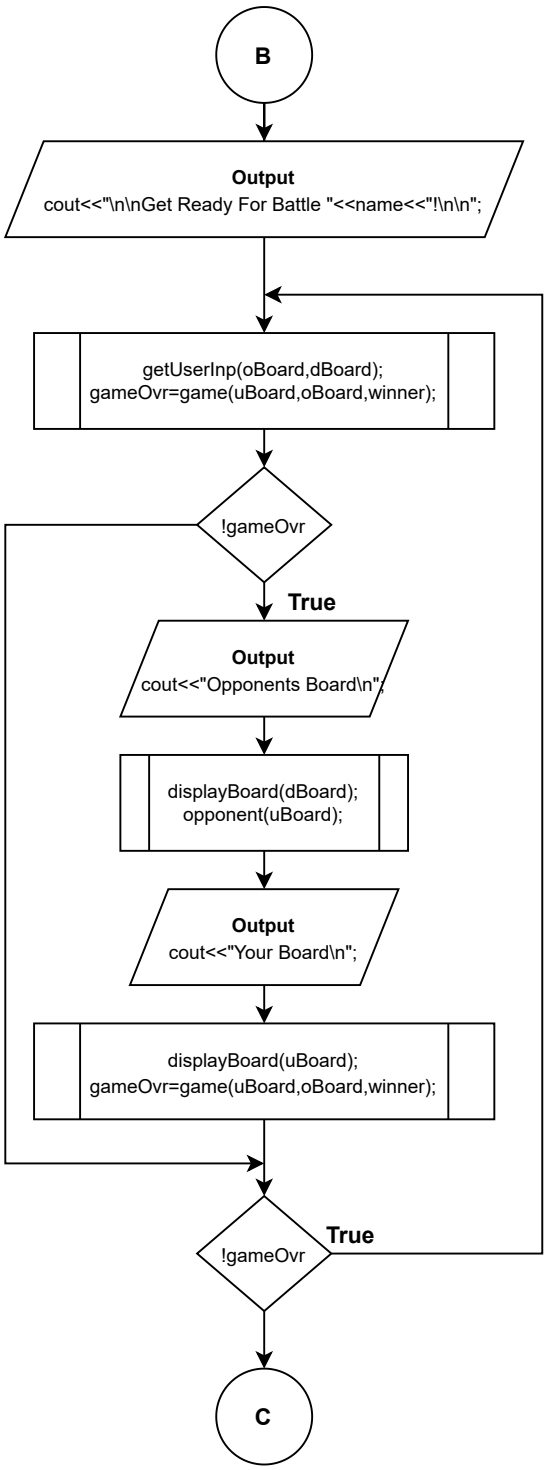
Pseudocode

Bring In System Libraries
Declare High Dimension Array Constant
Declare Function Prototypes
Enter Main Then,
    Set Random Number Seed
    Declare Variables and Initialize Game Boards



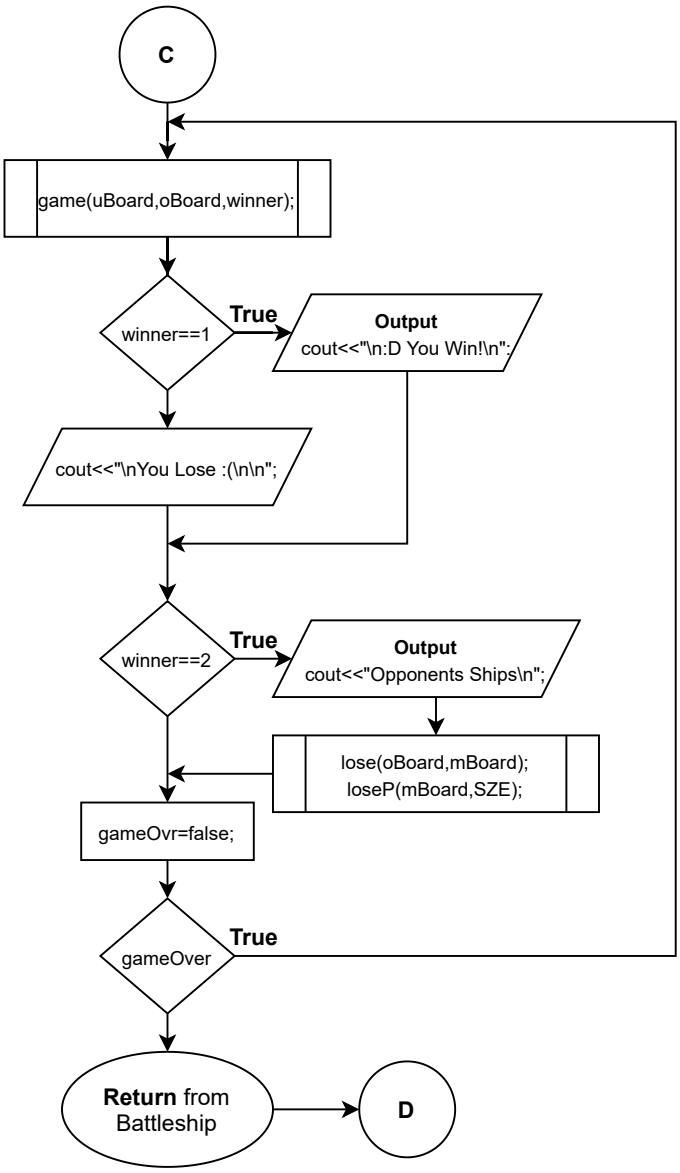
Generate Opponent's Ships
Generate User's Ships
Output Name of Game,
With Top and Bottom Boarder
Input User's Name

Do,
Display User's Board
Input Choice
Enter 1 To Generate New Battleship Locations
Enter 2 To Exit Program
Enter Anything Else To Continue
While Choice Is 1

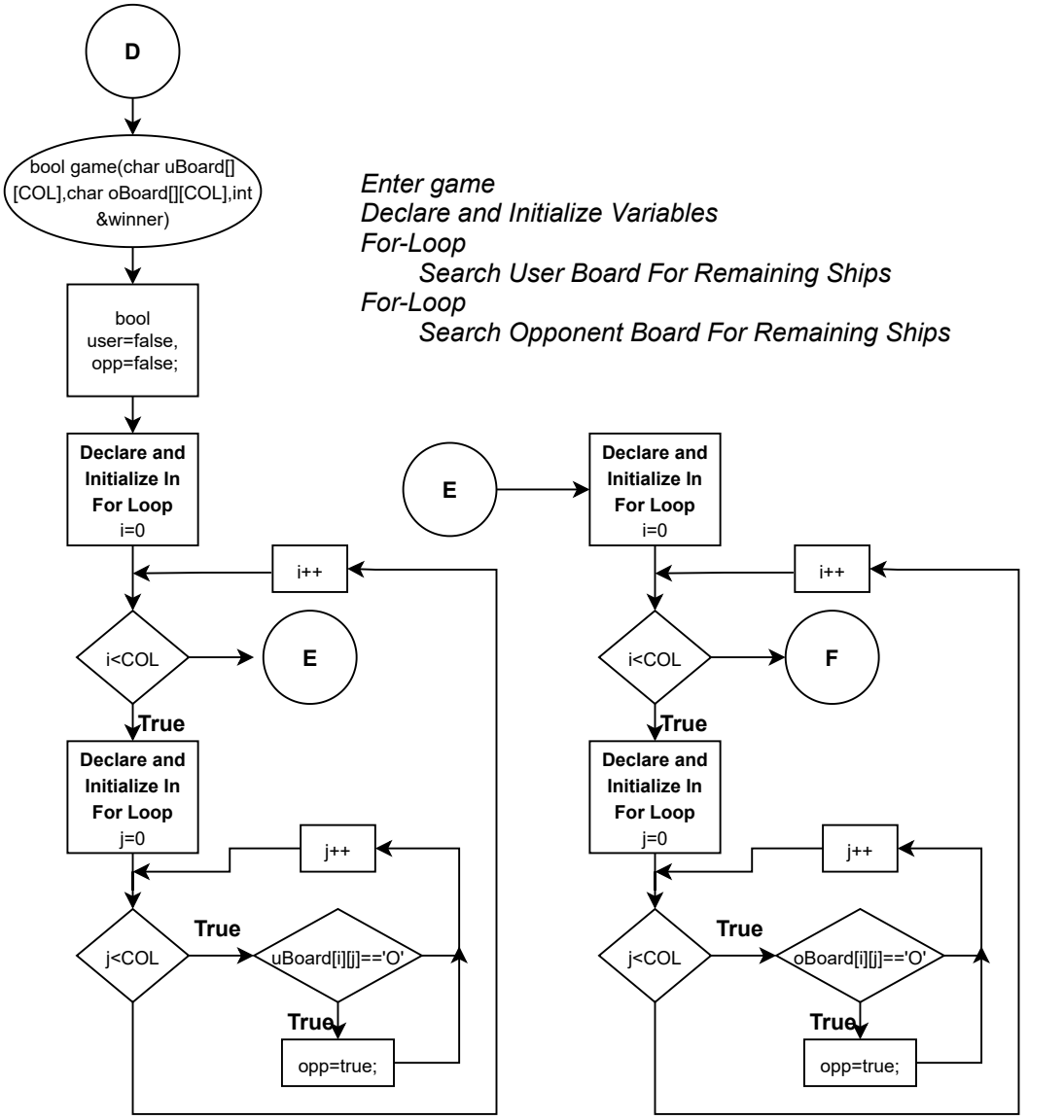


Output Message

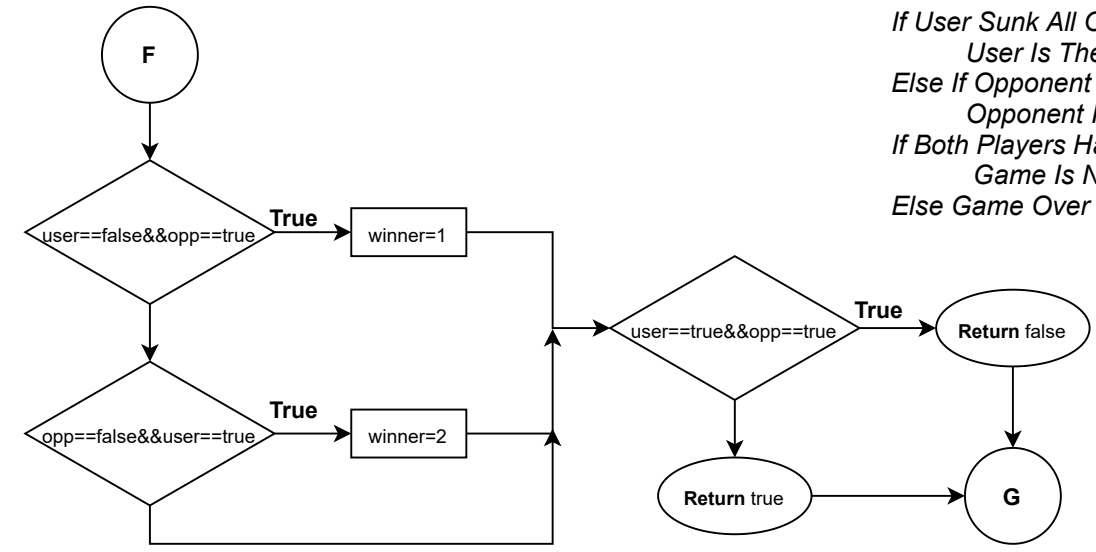
Do
Get User Input,
Input Row and Column To Attack
Check If Game Is Over
If Game Is Not Over
Display Opponents Board
Opponent's Turn To Attack
Display User's Board
Check If Game Is Over
While Game Is Not Over



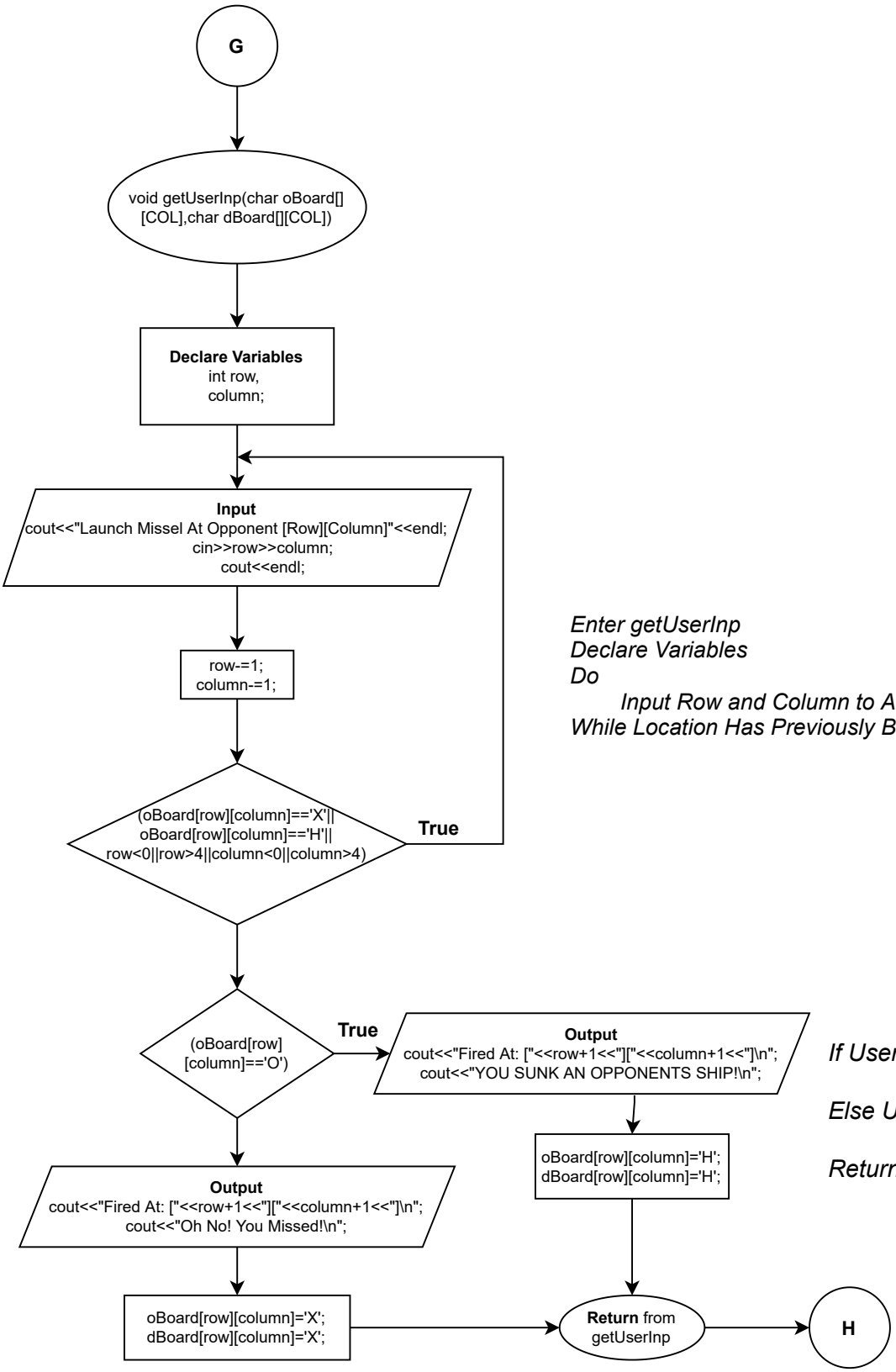
While Game Is Over
Check Who Won
If User Wins
Output Message
Else Output Message
If User Wins
Calculate and Output Score
Output Number Of Misses
If Opponent Wins
Output Mini Board Revealing Opponent Ships
Set GameOvr To False To Exit Loop



Enter game
Declare and Initialize Variables
For-Loop
Search User Board For Remaining Ships
For-Loop
Search Opponent Board For Remaining Ships

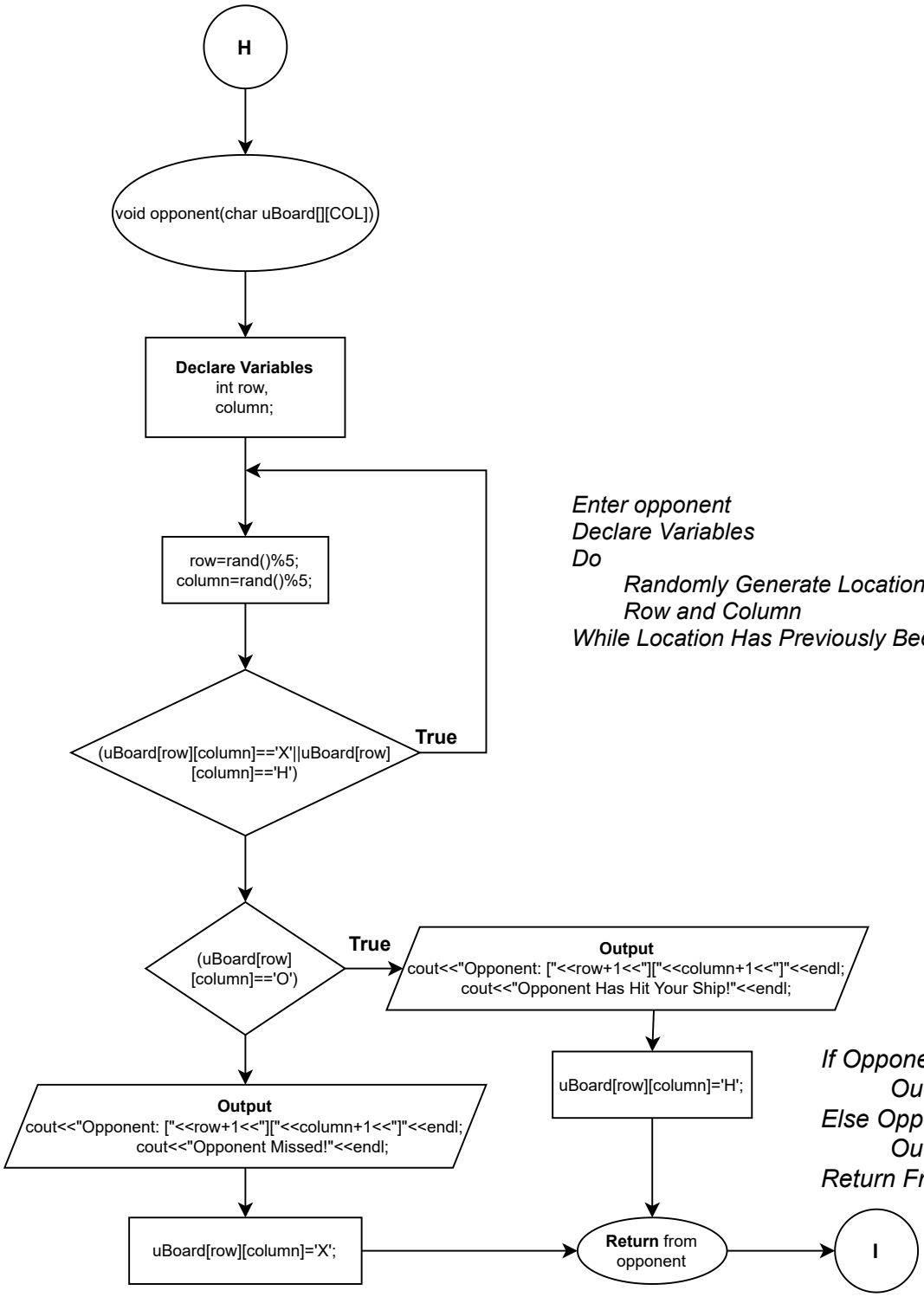


If User Sunk All Opponent Ships
User Is The Winner
Else If Opponent Sunk All User Ships
Opponent Is The Winner
If Both Players Have Ships Remaining
Game Is Not Over
Else Game Over



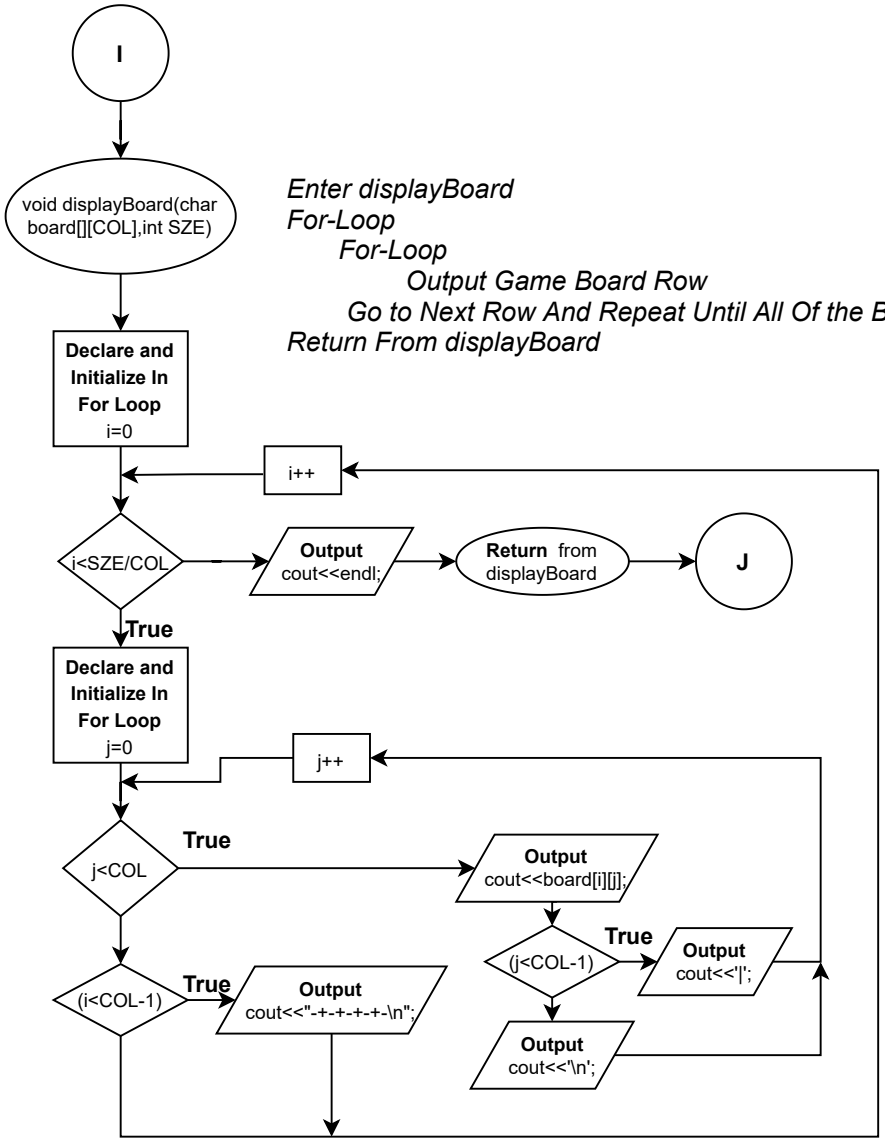
Enter getUserInp
Declare Variables
Do
Input Row and Column to Attack
While Location Has Previously Been Attacked

If User Hit Opponents Ship
Output Message and Set Location to 'H'
Else User Missed
Output Message and Set Location to 'X'
Return From getUserInp

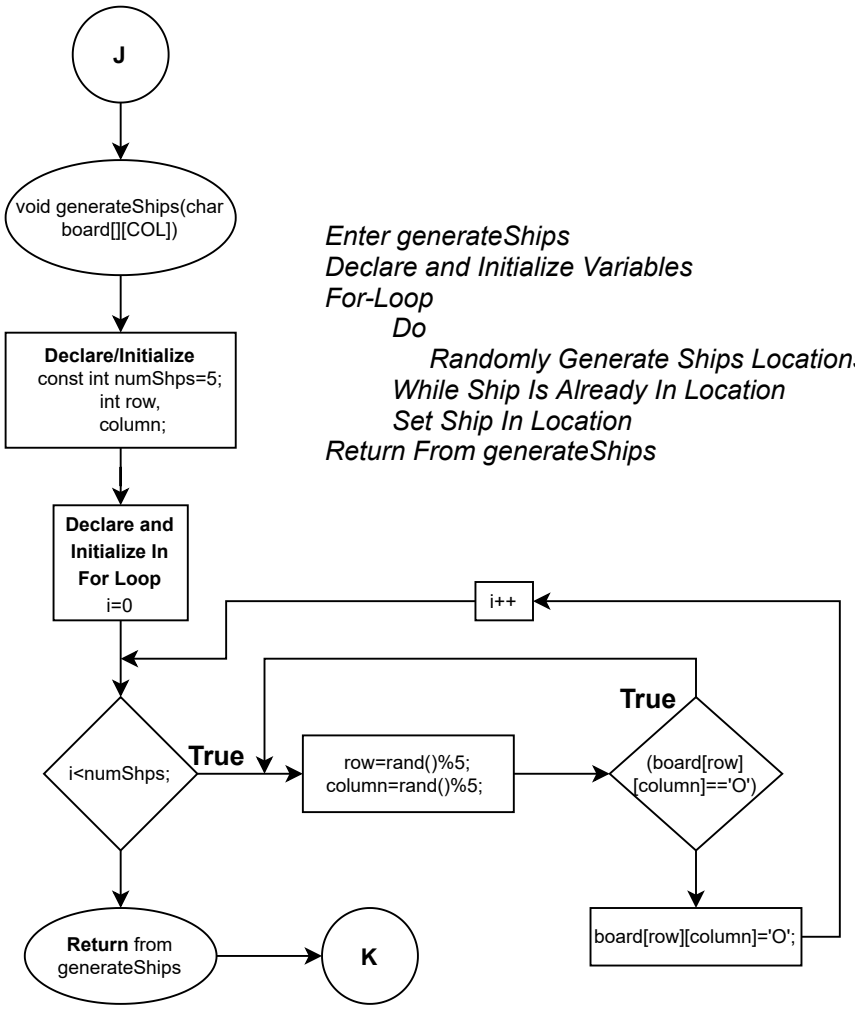


Enter opponent
Declare Variables
Do
    Randomly Generate Location To Attack
    Row and Column
While Location Has Previously Been Attacked

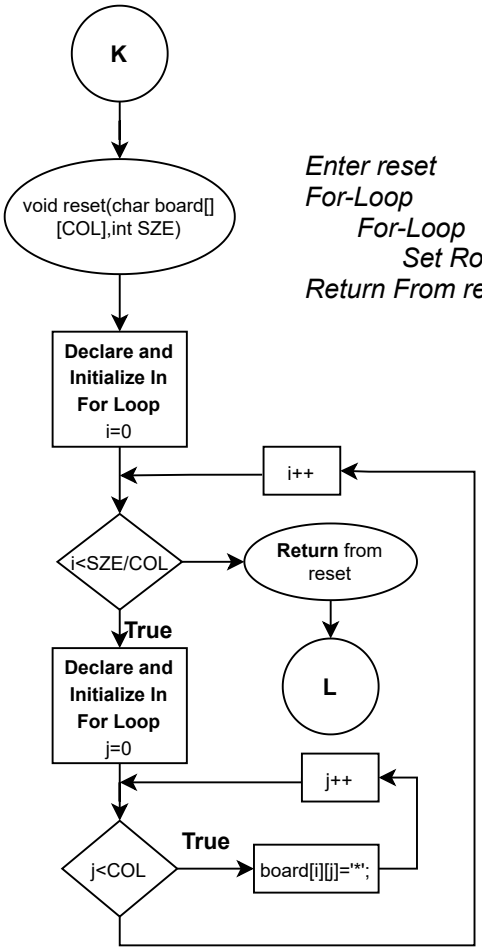
If Opponent Hit Users Ship
    Output Message and Set Location to 'H'
Else Opponent Missed
    Output Message and Set Location to 'X'
Return From opponent



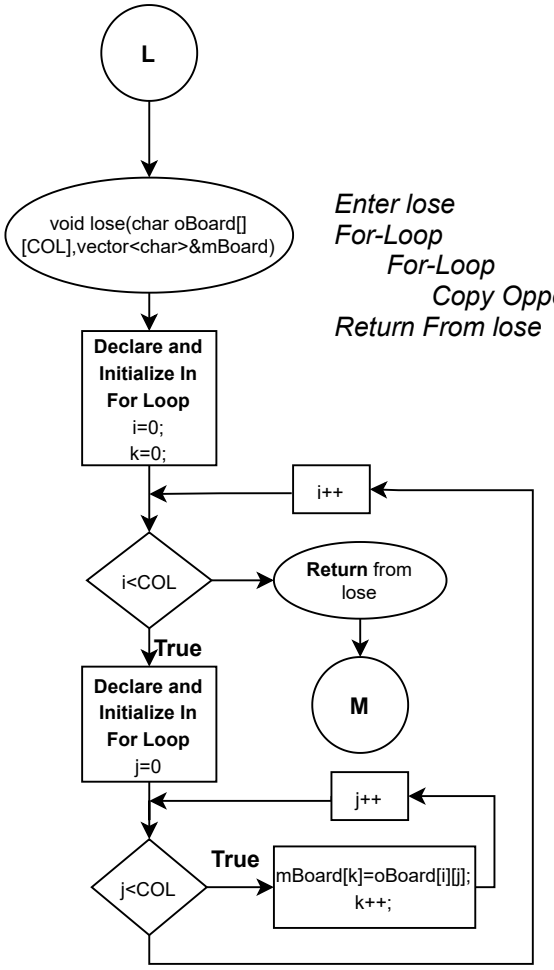
Enter displayBoard
For-Loop
For-Loop
Output Game Board Row
Go to Next Row And Repeat Until All Of the Board Is Outputed
Return From displayBoard



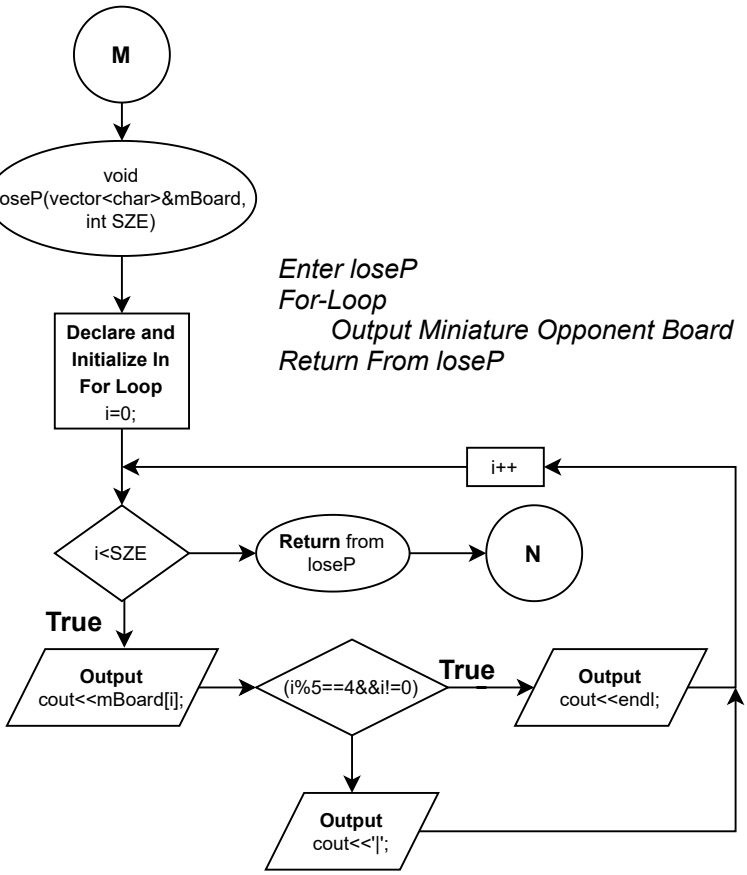
Enter generateShips
Declare and Initialize Variables
For-Loop
Do
Randomly Generate Ships Locations, Row and Column
While Ship Is Already In Location
Set Ship In Location
Return From generateShips



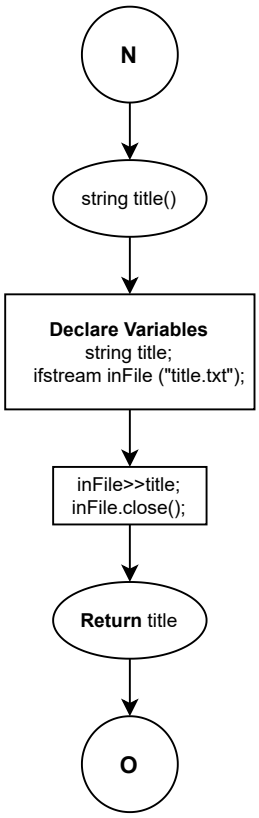
Enter reset
For-Loop
For-Loop
Set Row and Column to ''*
Return From reset



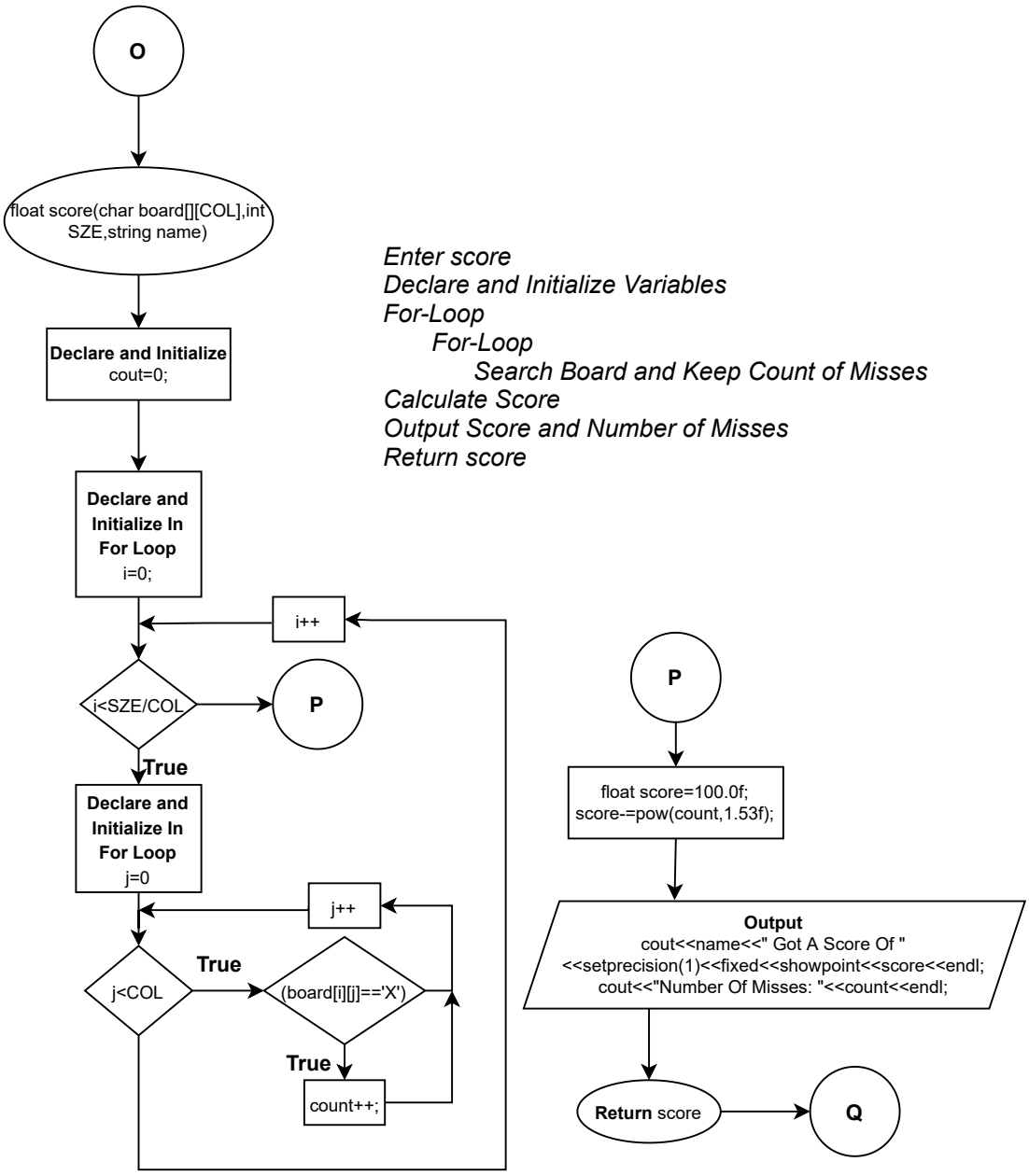
Enter lose
For-Loop
For-Loop
Copy Opponents Board to Vector
Return From lose



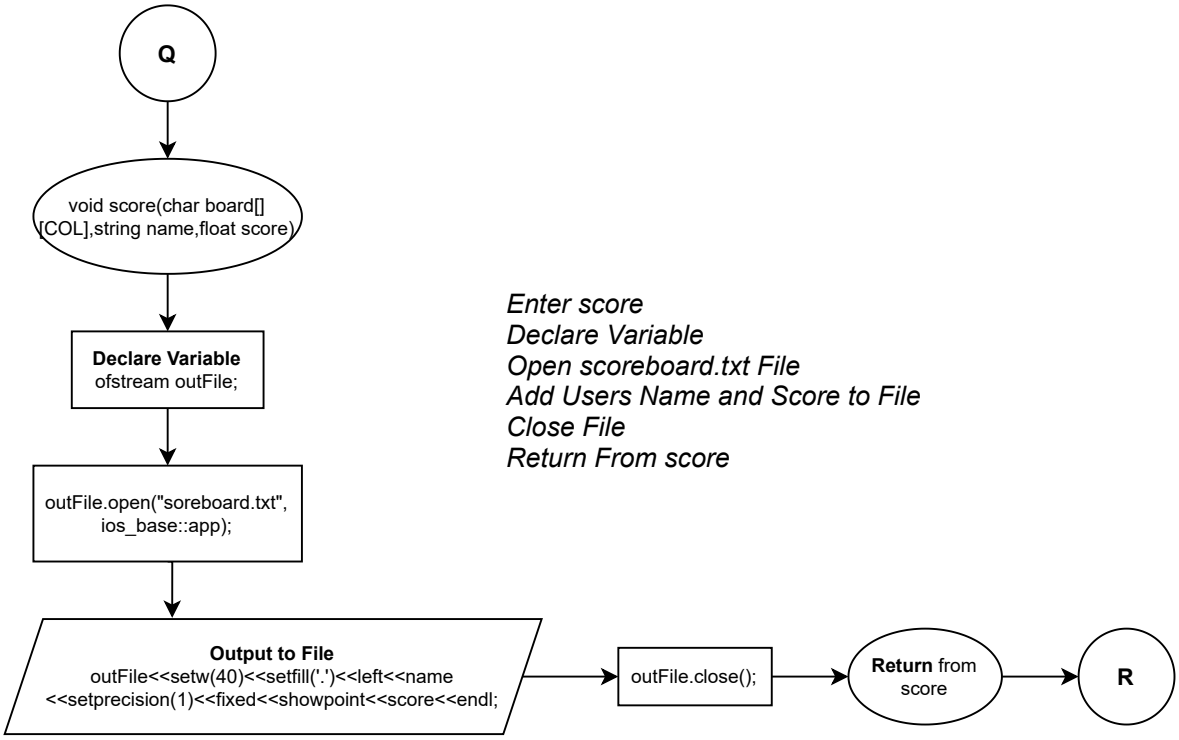
Enter loseP
For-Loop
Output Miniature Opponent Board
Return From loseP



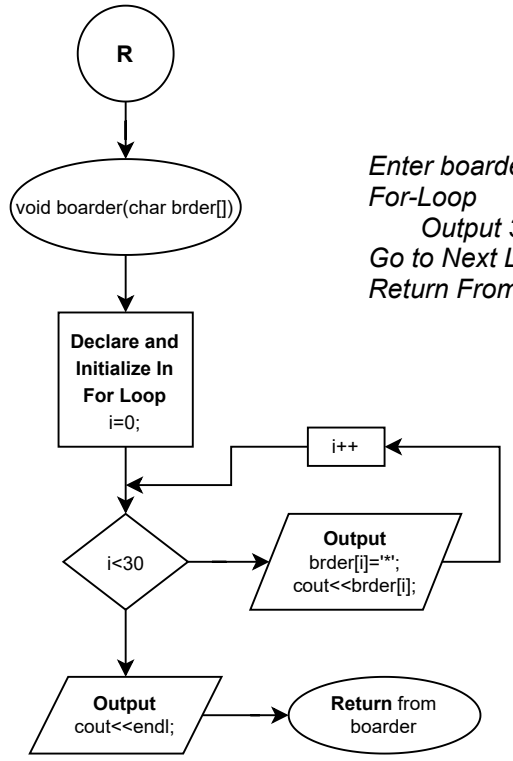
Enter title
Declare Variables
Open title.txt File
Read File and Store In title Variable
Return title



Enter score
Declare and Initialize Variables
For-Loop
For-Loop
Search Board and Keep Count of Misses
Calculate Score
Output Score and Number of Misses
Return score



Enter score
Declare Variable
Open scoreboard.txt File
Add Users Name and Score to File
Close File
Return From score



Enter boarder
For-Loop
Output 30 '' Characters*
Go to Next Line
Return From boarder