

---

```

clc
clear
fprintf('\n*****\n')
fprintf('*   Name:   Team Q   Date:   11/6/2020       *\n')
fprintf('*   Seat/Table:   00   File: Battleship.m *\n')
fprintf('*   Instructor: Jessica Thomas                *\n')
fprintf('*****\n')

*****
*   Name:   Team Q   Date:   11/6/2020       *
*   Seat/Table:   00   File: Battleship.m *
*   Instructor: Jessica Thomas                *
*****

clc
clear

% Initialize scene
my_scene = simpleGameEngine('Battleship.png',84,84);

% Set up variables to name the various sprites
blank_sprite = 1;
water_sprite = 2;
hit_sprite = 9;
miss_sprite = 10;

% Display empty board
board_display = water_sprite * ones(10,21);
board_display(:,11) = blank_sprite;
drawScene(my_scene,board_display)

% Set up hits and misses layer
hitmiss_display = blank_sprite * ones(10,21);

%set up random cpu board
cpu_ships = Setup();
%set up player board by letting them place ships
[player_ships, board_display, my_scene] = placeShip(board_display,
    my_scene);

%set variables needed for methods
lastHit = false;
confirmedHit = false;
count = 1;
cpuTurn =true;

numOfTurns = 0;

%while there is no winner
while((checkWinner(cpu_ships) == false)&&(checkWinner(player_ships) ==
    false))

```

---

---

```

    %if the cpu had its turn
    if(cpuTurn==true)
        %playerTurn
        if(numOfTurns~=10)&&(numOfTurns~=15)
            [~, hitmiss_display, cpu_ships] = playerTurn(my_scene,
cpu_ships, hitmiss_display, hit_sprite, miss_sprite);

            elseif(numOfTurns==10)
                fprintf("Use your 3x3 power-up!\n");
                [cpu_ships, hitmiss_display]=threeBy3(cpu_ships, my_scene,
hitmiss_display, hit_sprite, miss_sprite);

            else
                fprintf("Use your double shot power-up!");
                [cpu_ships, hitmiss_display] = twoTurns( my_scene,
cpu_ships, hitmiss_display, hit_sprite, miss_sprite, board_display);
            end

        numOfTurns = numOfTurns+1;
        %now its the cpus turn
        cpuTurn = false;

    end
    %check the player didn't win already before the cpu takes a turn
    if(checkWinner(player_ships)==true)
        fprintf("Player 1 wins!");
        break;
    end

    %if the cpu missed its last shot
    if(lastHit == false)
        %take a random guess
        cpu_row = randi([1 10]);
        cpu_col = randi([1 10]);
        %while the guess hasn't already been shot before
        while(player_ships(cpu_row, cpu_col)==-1)
            cpu_row = randi([1 10]);
            cpu_col = randi([1 10]);
        end
        %cpu turn
        [lastHit, hitmiss_display, player_ships] =
hitOrMiss(player_ships, hitmiss_display, cpu_row, cpu_col,
hit_sprite, miss_sprite);
        cpuTurn = true;

        %if the cpu has two hits in a row (has found a ship and the
        %direction it has been placed)
    elseif(confirmedHit ==true)
        %if the direction of the two hits is vertical

```

---

---

```

        if(((abs(move)==1)&&((cpu_row+move<11)&&(cpu_row
+move>0)))&&(player_ships(cpu_row+move, cpu_col)~= -1))
            %either move up or down a row (continue in the same
direction)
            cpu_row = cpu_row+move;
            %cpu turn
            [confirmedHit, hitmiss_display, player_ships] =
hitOrMiss(player_ships, hitmiss_display, cpu_row, cpu_col,
hit_sprite, miss_sprite);
            cpuTurn = true;
            %if the hits are horizontal
            elseif(((abs(move)==2)&&((cpu_col+move<11)&&(cpu_col
+move>0)))&&(player_ships(cpu_row, cpu_col+(move*.5))~= -1))
                %continue horizontally in the same direction
                cpu_col = cpu_col+(move*.5);
                %cpu turn
                [confirmedHit, hitmiss_display, player_ships] =
hitOrMiss(player_ships, hitmiss_display, cpu_row, cpu_col,
hit_sprite, miss_sprite);
                cpuTurn = true;
            else
                confirmedHit=false;
            end

            if(confirmedHit==false)
                lastHit=false;
            end

        %if lastHit is true and confirmed hit is false
        elseif((lastHit==true)&&(confirmedHit==false))
            % if the row number is within the bounds (10), and this spot
hasn't
            %been checked already
            if(((cpu_row+1<11)&&(player_ships(cpu_row+1, cpu_col)~= -1)))
                %have cpu check a row down from the last hit
                %cpu turn
                [confirmedHit, hitmiss_display, player_ships] =
hitOrMiss(player_ships, hitmiss_display, (cpu_row+1), cpu_col,
hit_sprite, miss_sprite);
                cpuTurn = true;
                %if this was also a hit
                if(confirmedHit ==true)
                    %this is two hits in a row
                    %move tells which way to continue checking to hit the
entire ship
                    move = 1;
                    cpu_row = cpu_row+1;
                end

                %if cpu_row is within the bounds (1-10) and this spot hasn't
been shot

```

---

---

```

    %already
    elseif((cpu_row-1>0)&&(player_ships(cpu_row-1, cpu_col)~= -1))

        %cpu turn
        [confirmedHit, hitmiss_display, player_ships] =
hitOrMiss(player_ships, hitmiss_display, (cpu_row-1), cpu_col,
hit_sprite, miss_sprite);
        cpuTurn = true;
        %if this was a hit
        if(confirmedHit ==true)
            %then there are two shots in a row and we now know the
            %ships direction
            %move is a variable that tells the ships direction and
            %which way to go to keep geussing
            move = -1;
            cpu_row = cpu_row-1;
        end

        %if cpu_col is within the bounds and this spot hasn't been
shot
        %before
        elseif(((cpu_col+1<11)&&(player_ships(cpu_row, cpu_col
+1)~= -1)))

            %cpu turn
            [confirmedHit, hitmiss_display, player_ships] =
hitOrMiss(player_ships, hitmiss_display, cpu_row, (cpu_col+1),
hit_sprite, miss_sprite);
            cpuTurn = true;
            %if this was a hit
            if(confirmedHit ==true)
                %then we know the ship is horizontal
                %move is a variable that tells which direction to
continue
                %looking
                move = 2;
                cpu_col = cpu_col+1;
            end

            %if the col is within the bounds (1-10) and this spot
hasn't
            %been guessed already
            elseif((cpu_col-1>0)&&(player_ships(cpu_row, cpu_col-1)~= -1))

                %cpu turn
                [confirmedHit, hitmiss_display, player_ships] =
hitOrMiss(player_ships, hitmiss_display, cpu_row, (cpu_col-1),
hit_sprite, miss_sprite);
                cpuTurn = true;
                %if this spot was a hit
                if(confirmedHit ==true)
                    %then the ship direction is horizontal

```

---

---

```

        %move is a variable that tells which direction to
continue
        %to look
        move = -2;
        cpu_col = cpu_col-1;
        %otherwise, go back to guessing randomly
    else

        confirmedHit = false;
        lastHit = false;
    end
    %otherwise, go back to guessing randomly
    else

        confirmedHit = false;
        lastHit = false;
    end
end

%update the scene the player is seeing
drawScene(my_scene,board_display,hitmiss_display)

end

```

*Error using ginput (line 84)  
Interrupted by figure deletion*

*Error in simpleGameEngine/getMouseInput (line 202)  
[X,Y,button] = ginput(1);*

*Error in placeShip (line 32)  
[row1,col1,~] = getMouseInput (my\_scene);*

*Error in BattleshipMain (line 34)  
[player\_ships, board\_display, my\_scene] = placeShip(board\_display,  
my\_scene);*

*Published with MATLAB® R2020b*