Pacman

**Built By Abdallah Abdelkader & Amr Ayoub**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

There is a lot of problems faced at this project as it is done from scratch without any tutorial or code resources

**The first one was the movement of pacman :**

We want to make the pacman at first smooth as much as possible and flexible so we set him to move by just three pixels:

And we added **4 keydown eventhandler**  to handle it ‘s movement across x and y coordinate

MoveLeft();

MoveRight();

MoveUp();

MoveDown();

There is a lot of problems here because we want him to move

In certain direction not to eat the wall or move beside the wall

We want pacman always move at multiplies of 27pixel

Because each image of each maze 27 pixel \* 27 pixel

So we have added Three functions two handle that part

function checkbroder(Currentx, Currenty, CurrentFace)

function CheckWall(DesireState)

function SetPixel(DesireState) {

checkborder check what is the next state without change it’s direction

checkwall and Setpixel both are handling what is pacman going to face if he changed his direction suddenly

if checkwall is not true then setpixel change his direction smoothly

with the condition that he should moving at multiplies of 27 pixel

**The Second one was the animation of pacman :**

@keyframes MoveRight {

    0% {

         background-image: url(resources/Right/pac1.png)

     }

     50% {

         background-image: url(resources/Right/pac2.png)

     }

     100% {

         background-image: url(resources/Right/pac3.png)

     }

 }

 @keyframes MoveLeft{

    0% {

         background-image: url(resources/Left/pac1.png)

     }

     50% {

         background-image: url(resources/Left/pac2.png)

     }

     100% {

         background-image: url(resources/Left/pac3.png)

     }

 }

 @keyframes MoveDown{

    0% {

         background-image: url(resources/Down/pac1.png)

     }

     50% {

         background-image: url(resources/Down/pac2.png)

     }

     100% {

         background-image: url(resources/Down/pac3.png)

     }

 }

 @keyframes MoveUp{

     0% {

         background-image: url(resources/Up/pac1.png)

     }

     50% {

         background-image: url(resources/Up/pac1.png)

     }

     100% {

         background-image: url(resources/Up/pac1.png)

     }

For each direction of movement there is an animation that change the background image of pacman at each direction

**The third one was the eating of fruit :**

 if (PacmanY % 27 == 0 && PacmanX % 27 == 0 && maze[row][col] == '.') {

            document.getElementById((PacmanY).toString() + (PacmanX).toString()).src = 'resources/' + 'x' + '.png';

each pic has an id of it ‘s position so if pacman move across certain position he changed the background of that image to black one

**The Fouth problem was the ghosts :**

each ghost is an object that has a property function of his random movement

var ghost\_1 = { element: document.getElementById('ghostOneOverlay') }

var ghost\_2 = { element: document.getElementById('ghostTwoOverlay') }

var ghost\_3 = { element: document.getElementById('ghostThreeOverlay') }

var ghost\_4 = { element: document.getElementById('ghostFourOverlay') }

ghost\_1.CrossRoad = function ()

and another property to check if the collision is happened between him and pacman then the gameover is displayed and the game start from the beginning

var ghostMoves = [

    function moveLeft(self) {

        var stepInterval = setInterval(function () {

            if ((Math.abs(this.x - PacmanX) < 27) && (Math.abs(this.y - PacmanY) < 27)) {

                showStatus('l')

                window.clearInterval(stepInterval)

            }