



INSTITUTO FEDERAL
SANTA CATARINA

Construindo um Jogo para a Web – *Pac-man*

Programação para a Internet

Prof. Vilson Heck Junior



Ministério
da Educação

Tecnologias Necessárias

- Tecnologias já Estudadas:
 - HTML;
 - CSS;
 - JavaScript;
- **Tecnologias Novas:**
 - **Computação Gráfica Básica;**
 - **Noções de Geometria;**
 - **Noções de Física;**
 - **Reprodução de Sons;**
 - **Enredo;**

Computação Gráfica

- É um campo da Ciência da Computação que estuda métodos para sintetizar e manipular digitalmente conteúdo visual:
 - Geração de imagens 2D;
 - Geração de imagens 3D (renderização);
 - Com ou sem animação;



Noções de Geometria

- Gráficos 2D ou 3D são na verdade a composição de pequenas peças geométricas:
- A relação espacial dada entre diferentes objetos existentes em uma cena deve ser respeitada:
 - Dois corpos não podem ocupar um mesmo lugar no espaço!

Noções de Física

- Objetos podem possuir algum tipo de movimento ou interação com outros objetos;
- Para isto, geralmente respeitam alguma(s) regras físicas:
 - Próximas a real: Simulação;
 - Diferenciadas: Arcade;

Reprodução de Sons

- O som é o elemento responsável por estimular o sentido da audição;
- Não tanto quanto os gráficos, mas os sons são responsáveis por completar uma boa sensação de imersão em jogos e entretenimento;
- Geralmente os sons (músicas ou barulhos) serão escolhidos conforme um determinado contexto ou acontecimento.



INSTITUTO FEDERAL
SANTA CATARINA

Enredo

- O enredo irá explicar ao usuário o que deverá ser feito e deve ser o principal responsável por atrair a atenção do jogador:
 - História;
 - Diversão;
 - Desafios;
 - Passatempo;
 - ...



Ministério
da Educação

Enredo

- Pac-man:

- Lançado em 22 de maio de 1980;
- Plataforma: arcade;
- Desenvolvido pela Namco e publicado pela Midway.



Atari -1982

Enredo

- Pac-man:

- Jogo cujo objetivo é comer todos os pontos em um labirinto;
- Os fantasmas não podem te alcançar;
- Existem pontos que te dão poder, depois de comê-los, por um certo tempo você poderá devorar os fantasmas.



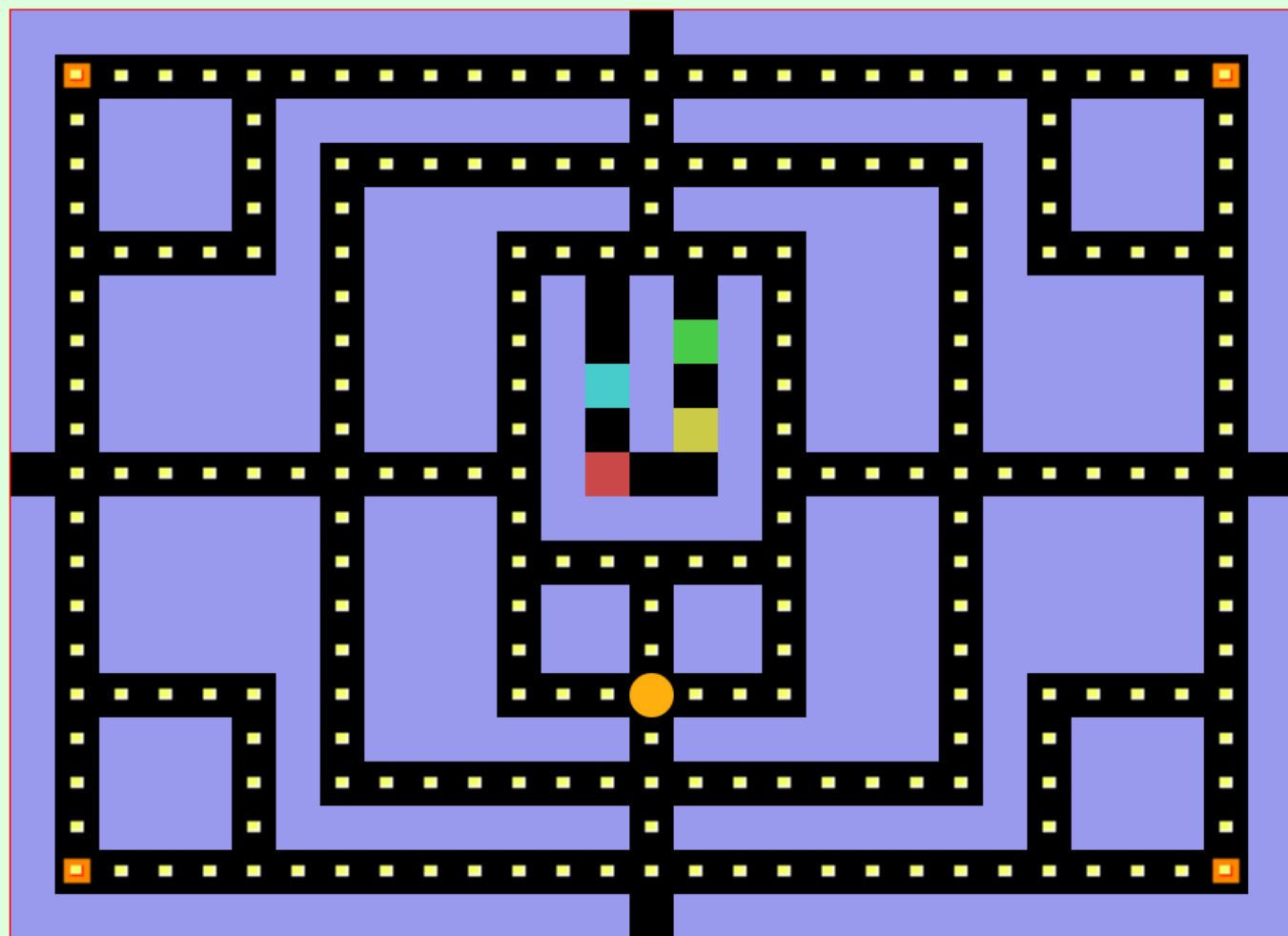


INSTITUTO FEDERAL
SANTA CATARINA

Nosso Conceito



Ministério
da Educação



Novo Jogo

Iniciar



INSTITUTO FEDERAL
SANTA CATARINA

Pac-man

LISTA DE RECURSOS INICIAIS



Ministério
da Educação



Recursos Iniciais

- Pasta: “Pacman”:
 - index.html
 - Construiremos de um documento web, inserindo todos os demais elementos necessários;
 - estilo.css
 - Definiremos algumas configurações de cores, bordas e outros para nossa interface;
 - pacman.js & configPacman.js & cenarioCriado.js
 - Faremos todo o processamento e configurações do jogo, ou seja, daremos vida aos elementos existentes no documento web.
 - Ghost.js
 - Implementaremos uma Classe Ghost para definir o comportamento dos fantasmas de pac-man;

index.html

- Crie o arquivo como **doctype** para **html 5**;
- Crie as tags para:
 - **<html>**, **<head>**, **<body>** e **<title>**;
- Estipule um **<link>** com arquivo de estilo;
- Adicione os arquivos de **<script>** dentro do **<head>**:
 - `cenarioCriado.js`
 - `configPacman.js`
 - `Ghost.js`
- Adicione o arquivo de **<script>** `pacman.js` **ao fim do <body>**;
 - *Importante: adicionar os outros arquivos js antes, pois o último precisa dos primeiros já executados.*

index.html

- Adicione os seguintes Tags com seus atributos dentro do <body>:
 - <div>
 - <canvas>Navegador não suportado!</canvas>
 - id = “tela” width=100 height=200
 - </div>
 - <div>
 - <button>Iniciar</button>
 - onclick=“pausar()” id=“btPausa”
 - <button>Novo Jogo</button>
 - onclick=“novoJogo()” id=“btNovo”
 - </div>



INSTITUTO FEDERAL
SANTA CATARINA

estilo.css

```
body { background-color: #DDFFDD;
        text-align: center; }

#tela {
    background-color: black;
    border: 1px solid red; }

button {
    background-color: #9999EE;
    color: white;
    border: 1px solid red;
    width: 100px;
    height: 30px;
    font-weight: bold;
    cursor: pointer; }

button:hover { background-color: #BBBBFF; }

button:disabled {
    background-color: #DDDDDD;
    cursor: not-allowed; }
```



INSTITUTO FEDERAL
SANTA CATARINA

Pac-man

DESENHANDO NO CANVAS



Ministério
da Educação





INSTITUTO FEDERAL
SANTA CATARINA

<canvas>

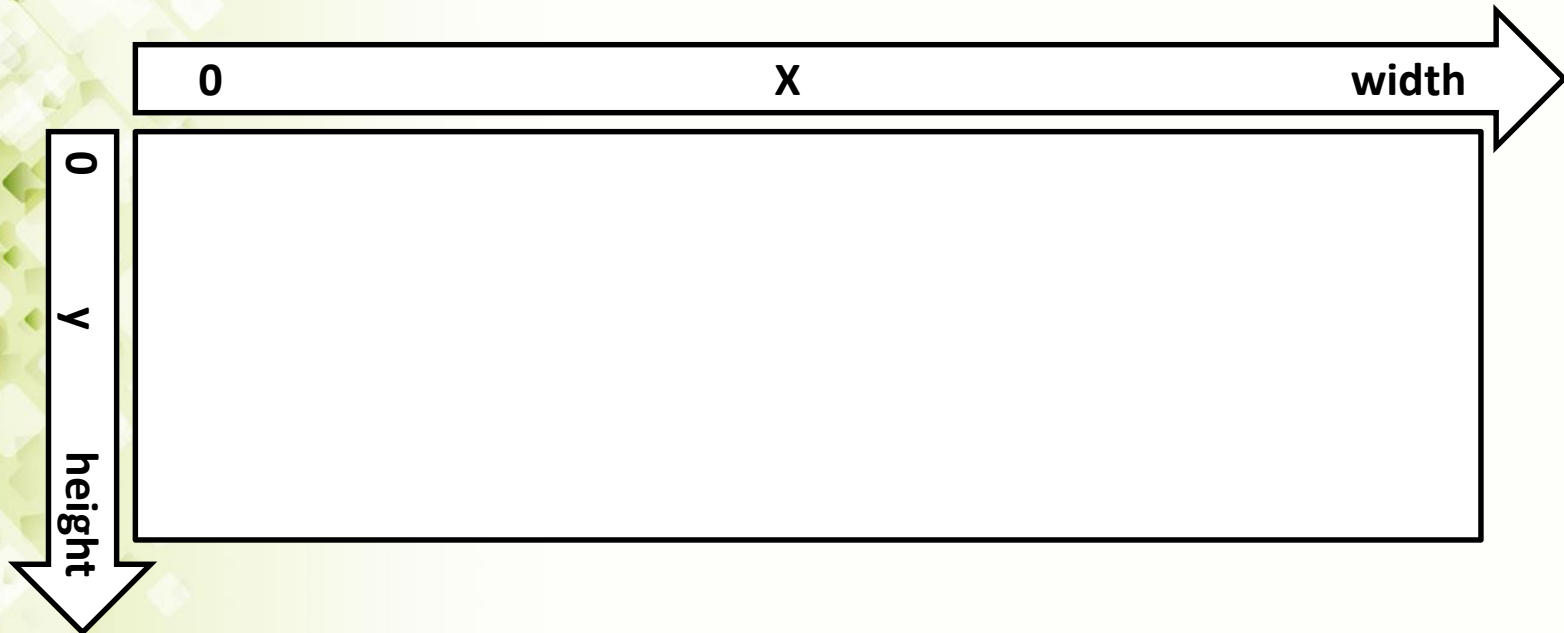
- Canvas é um termo inglês dado a alguns tipos de tela para pintura;
- No nosso caso, será uma área dentro do documento HTML onde poderemos pintar o que precisarmos;
- Nosso pincel e paleta de cores estão disponíveis através de código JavaScript.



Ministério
da Educação

<canvas>

- O Canvas é feito para oferecer suporte a rápido desenho de cenas bidimensionais ou tridimensionais:
 - Geralmente acelerado por Hardware;



pacman.js

//Recuperando referência dos objetos no documento

var canvas = document.getElementById("tela");

var ctx = canvas.getContext("2d");

var btPausa = document.getElementById("btPausa");

var btNovo = document.getElementById("btNovo");

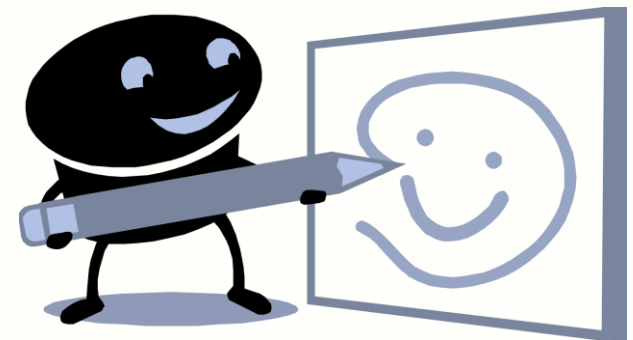
//Um pequeno teste (remover depois de testar)

ctx.fillStyle = "#FF0000"; //Usar cor vermelha

ctx.fillRect(20, 30, 50, 100); //x=20, y=30, w=50 e h=100

Desenhando

- Temos uma tela para desenho;
- Conhecemos uma primeira ferramenta para pintar algo;
- Temos que utilizar esta ferramenta de forma a construir o cenário inicial do nosso jogo;

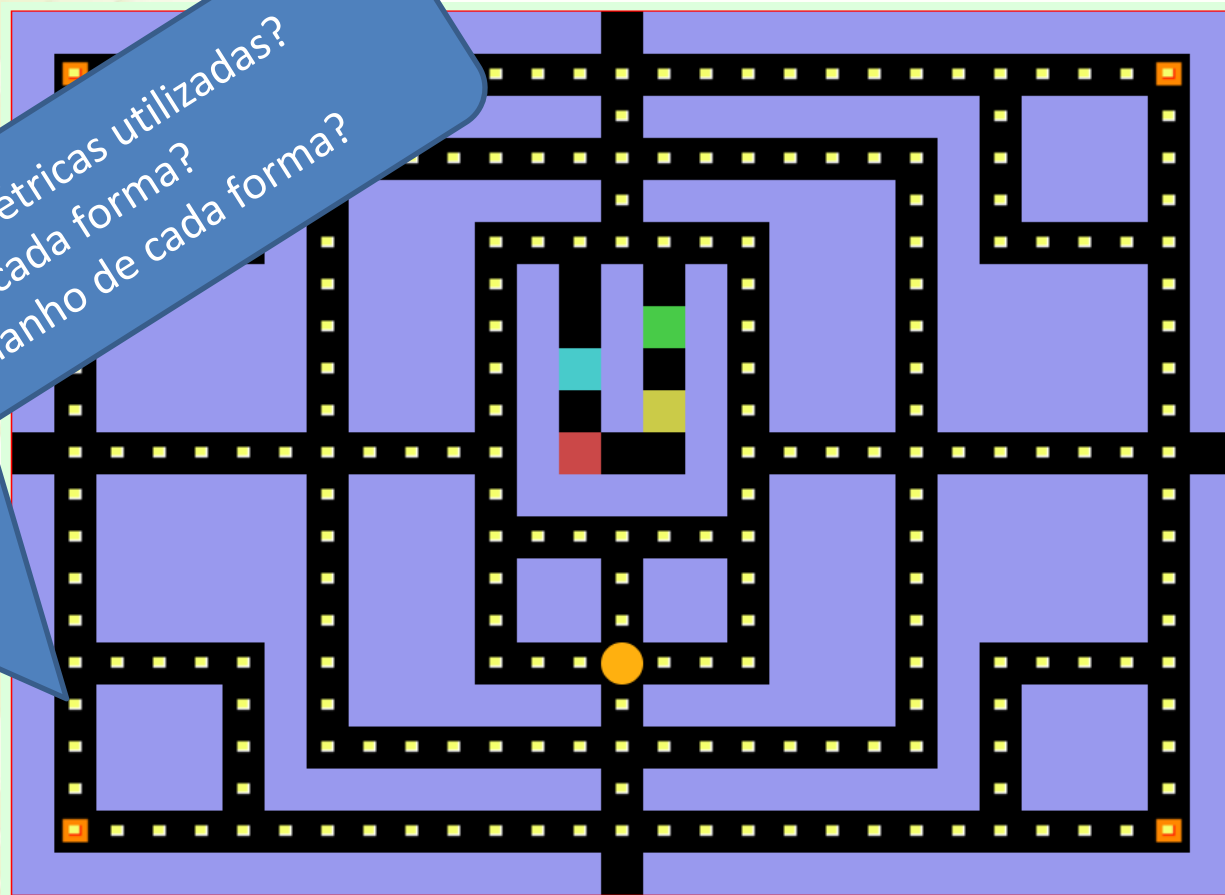




INSTITUTO FEDERAL
SANTA CATARINA

Relembrando

- Quais são as formas geométricas utilizadas?
- Qual é a quantidade de cada forma?
- Qual é a posição e tamanho de cada forma?



Novo Jogo

Iniciar

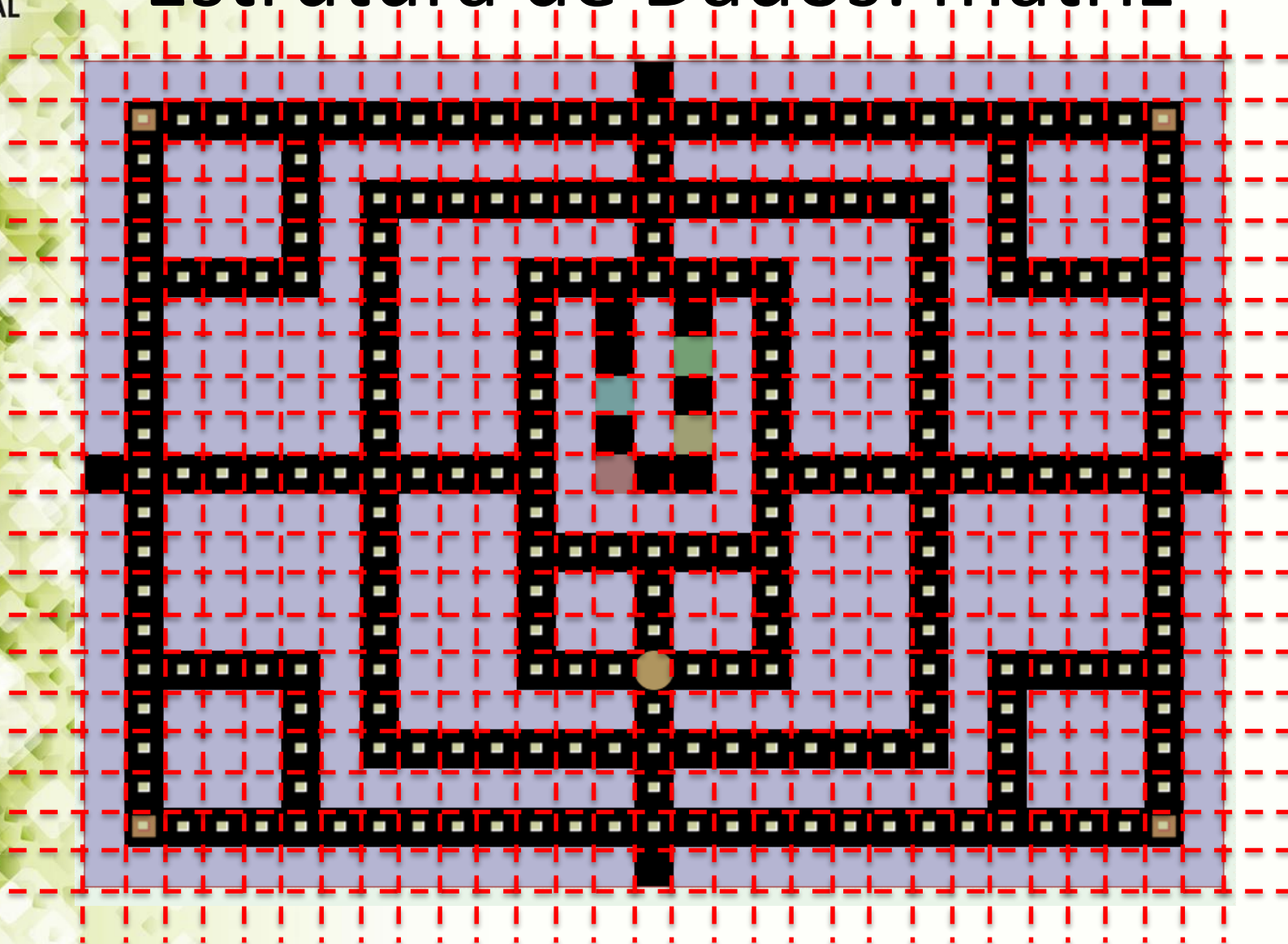


Ministério
da Educação



INSTITUTO FEDERAL
SANTA CATARINA

Estrutura de Dados: matriz



Ministério
da Educação

Posições da matriz

Descrição	Representação Gráfica	Código
Vazio		0 - (preto)
Parede		1 - (branco)
Poder		2 - (vermelho)
Ghost		3 - (verde)
Pac-man		4 - (azul)
Ponto		5 - (cinza)



INSTITUTO FEDERAL
SANTA CATARINA

cenarioCriado.js

Exemplo de cenário com 5 colunas e 6 linhas:

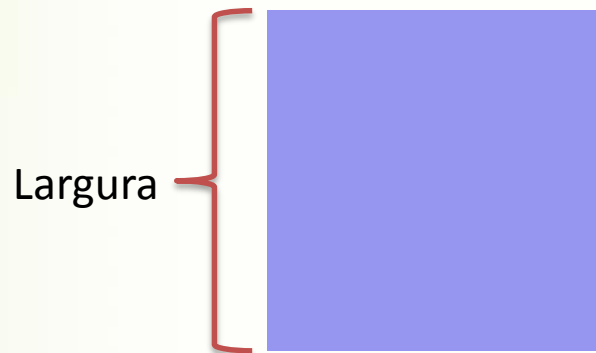
```
var cenárioCriado =  
[ [1, 1, 1, 1, 1],  
  [1, 0, 3, 0, 1],  
  [1, 0, 1, 0, 1],  
  [1, 0, 1, 0, 1],  
  [1, 0, 4, 0, 1],  
  [1, 1, 1, 1, 1] ];
```



Ministério
da Educação

Dimensões

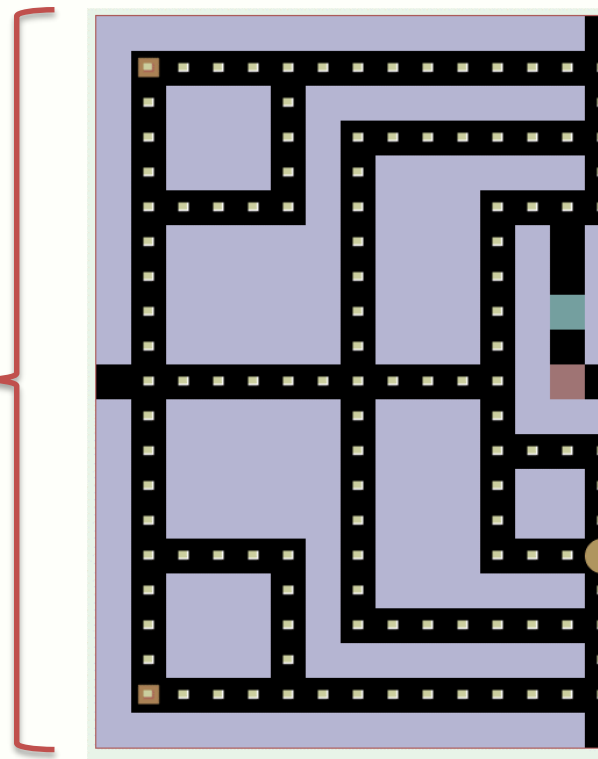
- Cada elemento da matriz terá uma largura e altura, sendo elementos quadrados: largura = altura.



Dimensões

- A largura, ou altura, geral do jogo, será definida pela largura de cada elemento x o número de elementos:

$$\text{AlturaTotal} = \text{Largura} * \text{nLinhas}$$





INSTITUTO FEDERAL
SANTA CATARINA

configPacman.js

```
var Cenario = function () {  
}  
Cenario.vazio = 0;  
Cenario.parede = 1;  
Cenario.poder = 2;  
Cenario.ghost = 3;  
Cenario.pacman = 4;  
Cenario.ponto = 5;  
//Armazena cópia do cenário  
Cenario.mapa = null;  
//Largura de cada elemento da matriz  
var largura = 30;
```



Ministério
da Educação



INSTITUTO FEDERAL
SANTA CATARINA

pacman.js

```
var nx = 0, ny = 0; //Número de colunas e linhas
function novoJogo() {
    Cenario.mapa = new Array();
    for (i = 0; i < cenarioCriado.length; i++) {
        Cenario.mapa.push(cenarioCriado[i].slice(0));
    }
    nx = Cenario.mapa[0].length;
    ny = Cenario.mapa.length;
    canvas.width = nx * largura;
    canvas.height = ny * largura;
    btPausa.disabled = false;
    btPausa.innerHTML = "Iniciar";
    desenharTudo();
}
```



Ministério
da Educação

pacman.js

- Entre em “Arquivos para Atividades Práticas” e baixe os seguintes arquivos para a pasta do seu projeto:
 - ponto.png
 - poder.png

//Imagens que serão desenhadas

```
var ponto = new Image();  
ponto.onload = desenharTudo;  
ponto.src = "ponto.png";
```

```
var poder = new Image();  
poder.onload = desenharTudo;  
poder.src = "poder.png";
```




INSTITUTO FEDERAL

SANTA CATARINA

pacman.js

```
function desenharTudo() {  
    //Limpar a tela  
    ctx.clearRect(0, 0, canvas.width, canvas.height);  
    //Cenário  
    ctx.fillStyle = "#9999EE";  
    for (y = 0; y < ny; y++) {  
        for (x = 0; x < nx; x++) {  
            if (Cenario.mapa[y][x] == Cenario.parede) {  
                ctx.fillRect(x * largura, y * largura, largura, largura);  
            } else if (Cenario.mapa[y][x] == Cenario.ponto) {  
                ctx.drawImage(ponto, x * largura, y * largura, largura,  
largura);  
            } else if (Cenario.mapa[y][x] == Cenario.poder) {  
                ctx.drawImage(poder, x * largura, y * largura, largura,  
largura);  
            } } //else if & for x  
        } } //for y & function
```



Ministério
da Educação

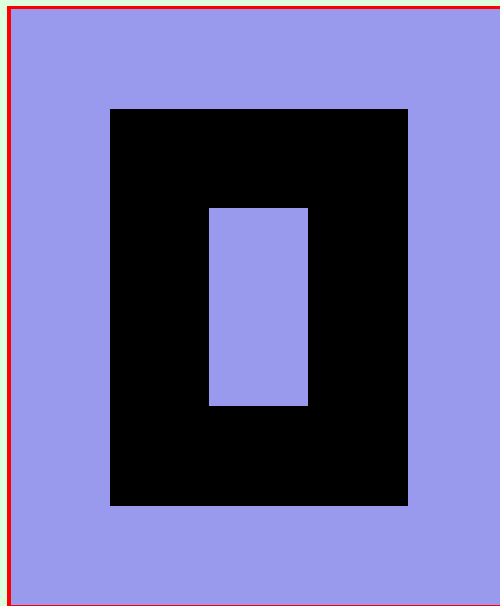
Pac-man

- Insira o seguinte comando ao fim do arquivo pacman.js:
 - **novoJogo();**
- Teste sua página!





INSTITUTO FEDERAL
SANTA CATARINA



Iniciar

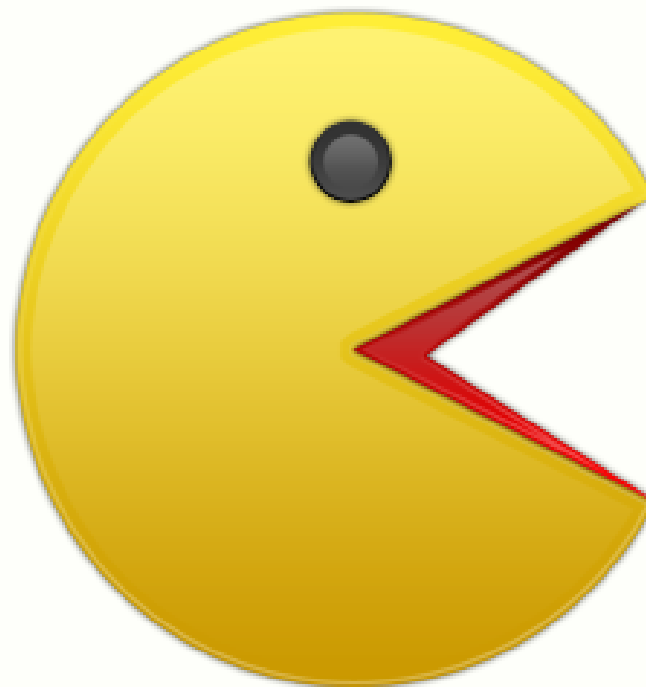
Novo Jogo



Ministério
da Educação



INSTITUTO FEDERAL
SANTA CATARINA



Pac-man

O ATOR PRINCIPAL



Ministério
da Educação

Pac-man

- O Pac-man será representado na memória por duas variáveis que indicam sua posição em x e y :
 - *var* px, py ;
- Quando estivermos carregando o cenário, iremos iniciar estas variáveis com a posição determinada para `Cenario.pacman`.



INSTITUTO FEDERAL
SANTA CATARINA

pacman.js

```
var px = -1, py = -1; //Posição do PAC-MAN
```

```
function novoJogo() {  
    canvas.height = ny * largura;  
  
    for (y = 0; y < ny; y++) {  
        for (x = 0; x < nx; x++) {  
            if (Cenario.mapa[y][x] == Cenario.pacman) {  
                px = x;  
                py = y;  
            }  
        }  
    }  
  
    btPausa.disabled = false;
```



Ministério
da Educação



INSTITUTO FEDERAL
SANTA CATARINA

pacman.js

```
function desenharTudo() {  
    //Cenário  
    //...  
  
    //Pacman  
    ctx.fillStyle = "#FFB00F";  
    ctx.beginPath();  
    ctx.arc(px * largura + (largura / 2), py * largura +  
    (largura / 2), largura / 2, Math.PI * 2, false);  
    ctx.closePath();  
    ctx.fill();  
}
```



Ministério
da Educação

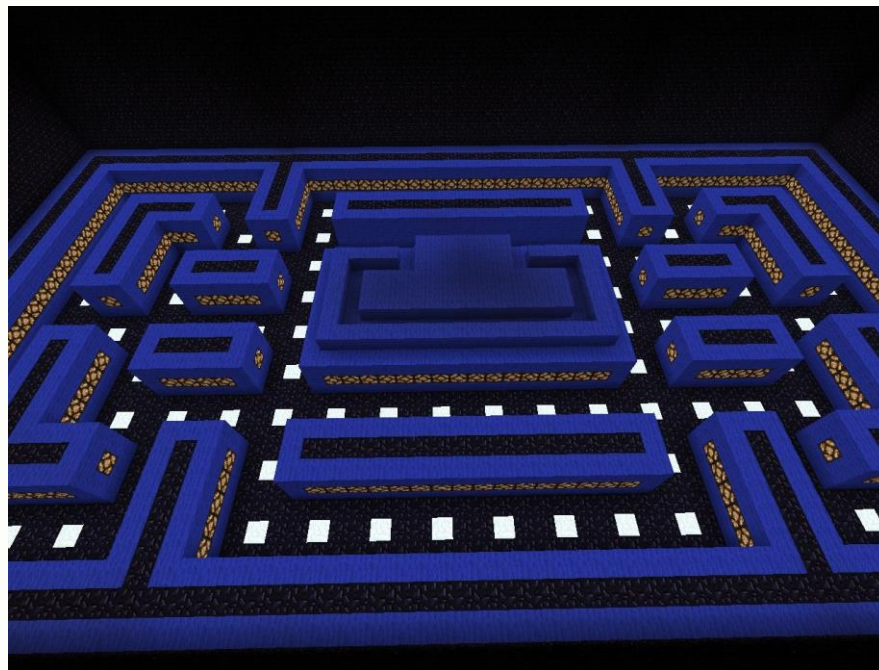
Pac-man

- Teste sua página!





INSTITUTO FEDERAL
SANTA CATARINA



Pac-man

GERAÇÃO DE MAPAS



Ministério
da Educação

Geração de Mapas

- Em “Arquivos para Atividades Práticas”
 - Na pasta “pacman”;
 - Baixe PacmanMapGenerator.zip;
 - Isto é um projeto NetBeans!
- Gerar um arquivo BMP no MS Paint!
 - Observar cores que irão definir objetos.
- Converter com o software Java!

Geração de Mapas

- Em “Arquivos para Atividades Práticas”
 - Na pasta “pacman”;
 - Baixe “cenario.png” para qualquer pasta
 - Converta o arquivo com o Gerador de Mapas baixado anteriormente;
 - salve o arquivo gerado com nome **cenarioCriado.js** ;
 - Coloque o arquivo gerado dentro da pasta do seu projeto, substituindo o arquivo digitado anteriormente.

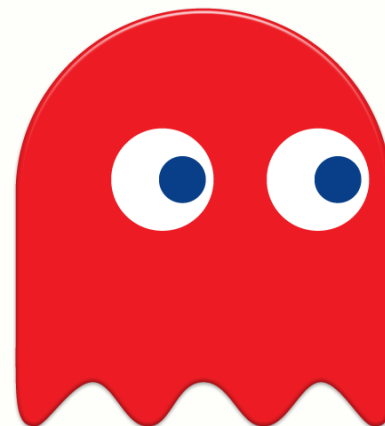
Pac-man

- Teste sua página!





INSTITUTO FEDERAL
SANTA CATARINA



Pac-man

OS VILÕES!



Ministério
da Educação

Os Vilões

- No Pac-man, os vilões são fantasmas que andam pelo labirinto;
- Em média, teremos quatro fantasmas.
- Em nossa implementação, cada fantasma será uma instância de uma classe chamada Ghost;
- Cada vez que encontrarmos o Cenario.ghost no mapa, iremos instanciar um novo fantasma.



INSTITUTO FEDERAL
SANTA CATARINA

configPacman.js

```
//Direcoes  
var Direcao = function(){  
}  
Direcao.naoDefinida = -1;  
Direcao.cima = 0;  
Direcao.baixo = 1;  
Direcao.esquerda = 2;  
Direcao.direita = 3;
```



Ministério
da Educação



INSTITUTO FEDERAL
SANTA CATARINA

Ghost.js

//Classe Ghost -> Define atributos e ações dos fantasmas

```
var Ghost = function(x, y, cor) {
```

Construtor

```
  //Atributos dinâmicos
```

```
  this.xi = x; //ponto x de inicio
```

```
  this.yi = y; //ponto y de inicio
```

```
  this.x = x;
```

```
  this.y = y;
```

```
  this.cor = cor;
```

```
  this.direcaoAtual = Direcao.naoDefinida;
```

```
  //Métodos dinâmicos
```

```
  this.desenhar = function(ct) {
```

```
    ct.fillStyle = this.cor;
```

```
    ct.fillRect(this.x * largura, this.y * largura,  
                largura, largura);
```



} Ministério
da Educação



INSTITUTO FEDERAL
SANTA CATARINA

Ghost.js

Parte “static”
Inserir fora do construtor

//Elementos estáticos da classe Ghost

```
Ghost.cores = new Array();  
Ghost.cores.push("rgba( 85, 238, 85, 0.85)");  
Ghost.cores.push("rgba( 85, 238, 238, 0.85)");  
Ghost.cores.push("rgba(238, 238, 85, 0.85)");  
Ghost.cores.push("rgba(238, 85, 85, 0.85)");  
Ghost.cores.push("rgba(238, 85, 238, 0.85)");
```



Ministério
da Educação



INSTITUTO FEDERAL
SANTA CATARINA

pacman.js

```
var ghosts = new Array(); //Armazena referencias dos Ghosts
function novoJogo() {
    ghosts.length = 0;
    var nGhosts = 0;
    for (y = 0; y < ny; y++) {
        for (x = 0; x < nx; x++) {
            if (Cenario.mapa[y][x] == Cenario.pacman) {
                ...
            }
            if (Cenario.mapa[y][x] == Cenario.ghost) {
                ghosts.push(new Ghost(x, y,
                                     Ghost.cores[nGhosts++]));
            }
        }
    }
}
```



Ministério
da Educação



INSTITUTO FEDERAL
SANTA CATARINA

pacman.js

```
function desenharTudo() {  
    //Cenário  
    //...  
  
    //Pacman  
    ...  
  
    //Fantasmas  
    for (i = 0; i < ghosts.length; i++) {  
        ghosts[i].desenhar(ctx);  
    }  
}
```



Ministério
da Educação

Pac-man

- Teste sua página!





INSTITUTO FEDERAL
SANTA CATARINA

Pac-man

COLOCANDO VIDA



Ministério
da Educação

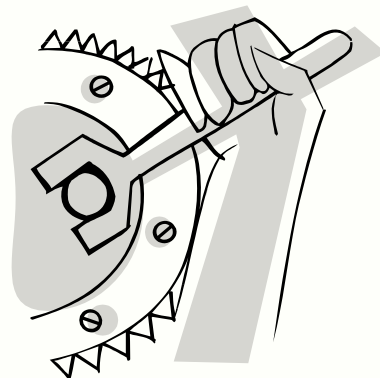


O que precisamos?

- Fazer o Pac-man se movimentar ao pressionar de tecla?
- Fazer os fantasmas se movimentarem:
 - Com qual intervalo de tempo?
 - Para qual direção?
- O que acontece se o usuário ordenar que o Pac-man se mova contra uma parede?
- O que ocorre quando o Pac-man encontra um ponto?
- O que ocorre quando o Pac-man encontra um ponto de poder?
- O que ocorre quando o Pac-man encontra um fantasma?
 - E se o pac-man tiver comido um ponto de poder?
- E quando acabarem todos os pontos do mapa?

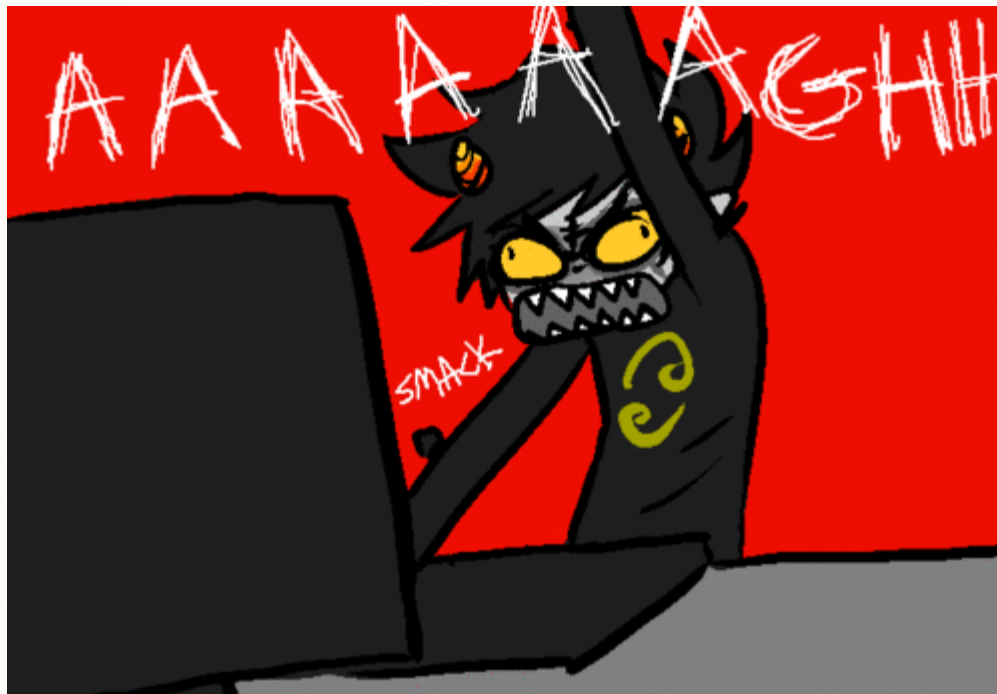
Movimentação do Pac-man

- O Pac-man irá se movimentar conforme teclas que o usuário pressionar:
 - Setas direcionais;
 - Outras teclas configuradas;





INSTITUTO FEDERAL
SANTA CATARINA



Pac-man

INTERAGINDO COM O USUÁRIO



Ministério
da Educação

Eventos!

- A interação é dada por uma troca entre a máquina e o usuário;
- A máquina fornece principalmente imagens que descrevem uma situação, onde pode ser necessária a intervenção do usuário;
- O usuário irá intervir basicamente através de comandos!
 - Comandos são captados através de eventos.

Eventos!

- Nosso **document** possui propriedades de eventos que podem ser associadas à funções quaisquer;
- Estas funções determinam algo a ser feito quando aquele evento ocorrer:
 - **document.onkeydown**
 - Ao descer uma tecla qualquer;
 - **document.onkeyup**
 - Ao soltar uma tecla qualquer;



pacman.js

```
document.onkeydown = onKD; //Eventos de tecla para método onKD
var setaCima = false;
var setaBaixo = false;
var setaEsquerda = false;
var setaDireita = false;
function onKD(evt) {
    if (evt.keyCode == Teclas.direita) {
        setaDireita = true;
    }
    if (evt.keyCode == Teclas.esquerda) {
        setaEsquerda = true;
    }
    if (evt.keyCode == Teclas.cima) {
        setaCima = true;
    }
    if (evt.keyCode == Teclas.baixo) {
        setaBaixo = true;
    }
}
```




INSTITUTO FEDERAL
SANTA CATARINA

configPacman.js

```
//Configurações das teclas  
var Teclas = function () {  
  }  
Teclas.cima = 38;  
Teclas.baixo = 40;  
Teclas.esquerda = 37;  
Teclas.direita = 39;
```



Ministério
da Educação



pacman.js

```
function moverPacman() {  
    if (setaDireita) {  
        setaDireita = false;  
        px++;  
    }  
    if (setaEsquerda) {  
        setaEsquerda = false;  
        px--;  
    }  
    if (setaCima) {  
        setaCima = false;  
        py--;  
    }  
    if (setaBaixo) {  
        setaBaixo = false;  
        py++;  
    }  
}
```

Pac-man

- Colocar temporariamente os comandos dentro, no fim da função onKD():
 - moverPacman();
 - desenharTudo();
- Teste sua página!





INSTITUTO FEDERAL
SANTA CATARINA



Pac-man

DETECTANDO COLISÕES



Ministério
da Educação

Detectando Colisões

- É indesejável que dois objetos ocupem o mesmo lugar no espaço ao mesmo tempo.
- Para isto, antes de aceitar a movimentação do Pac-man, precisamos verificar se a posição para onde ele irá se mover está disponível.



pacman.js

```
function moverPacman() {  
  
    if (setaDireita) {  
        setaDireita = false;  
        if (px + 1 < nx) {  
            if (Cenario.mapa[py][px + 1] != Cenario.parede) {  
                px++;  
            }  
        } else if (Cenario.mapa[py][0] != Cenario.parede) {  
            px = 0;  
        }  
    }  
  
    ...  
  
    ...  
  
    ...  
}
```



pacman.js

```
function moverPacman() {  
  
    ...  
  
    if (setaEsquerda) {  
        setaEsquerda = false;  
        if (px - 1 >= 0) {  
            if (Cenario.mapa[py][px - 1] != Cenario.parede) {  
                px--;  
            }  
        } else if (Cenario.mapa[py][nx - 1] != Cenario.parede) {  
            px = nx - 1;  
        }  
    }  
  
    ...  
  
    ...  
}
```



pacman.js

```
function moverPacman() {  
  
    ...  
  
    ...  
    if (setaCima) {  
        setaCima = false;  
        if (py - 1 >= 0) {  
            if (Cenario.mapa[py - 1][px] != Cenario.parede) {  
                py--;  
            }  
        } else if (Cenario.mapa[ny - 1][px] != Cenario.parede) {  
            py = ny - 1;  
        }  
    }  
  
    ...  
}
```




pacman.js

```
function moverPacman() {  
  
    ...  
  
    ...  
  
    ...  
  
    if (setaBaixo) {  
        setaBaixo= false;  
        if (py + 1 < ny) {  
            if (Cenario.mapa[py + 1][px] != Cenario.parede) {  
                py++;  
            }  
        } else if (Cenario.mapa[0][px] != Cenario.parede) {  
            py = 0;  
        }  
    }  
}
```

Pac-man

- Colocar temporariamente os comandos dentro, no fim da função onKD():
 - moverPacman();
 - desenharTudo();
- Teste sua página!





INSTITUTO FEDERAL
SANTA CATARINA



Pac-man

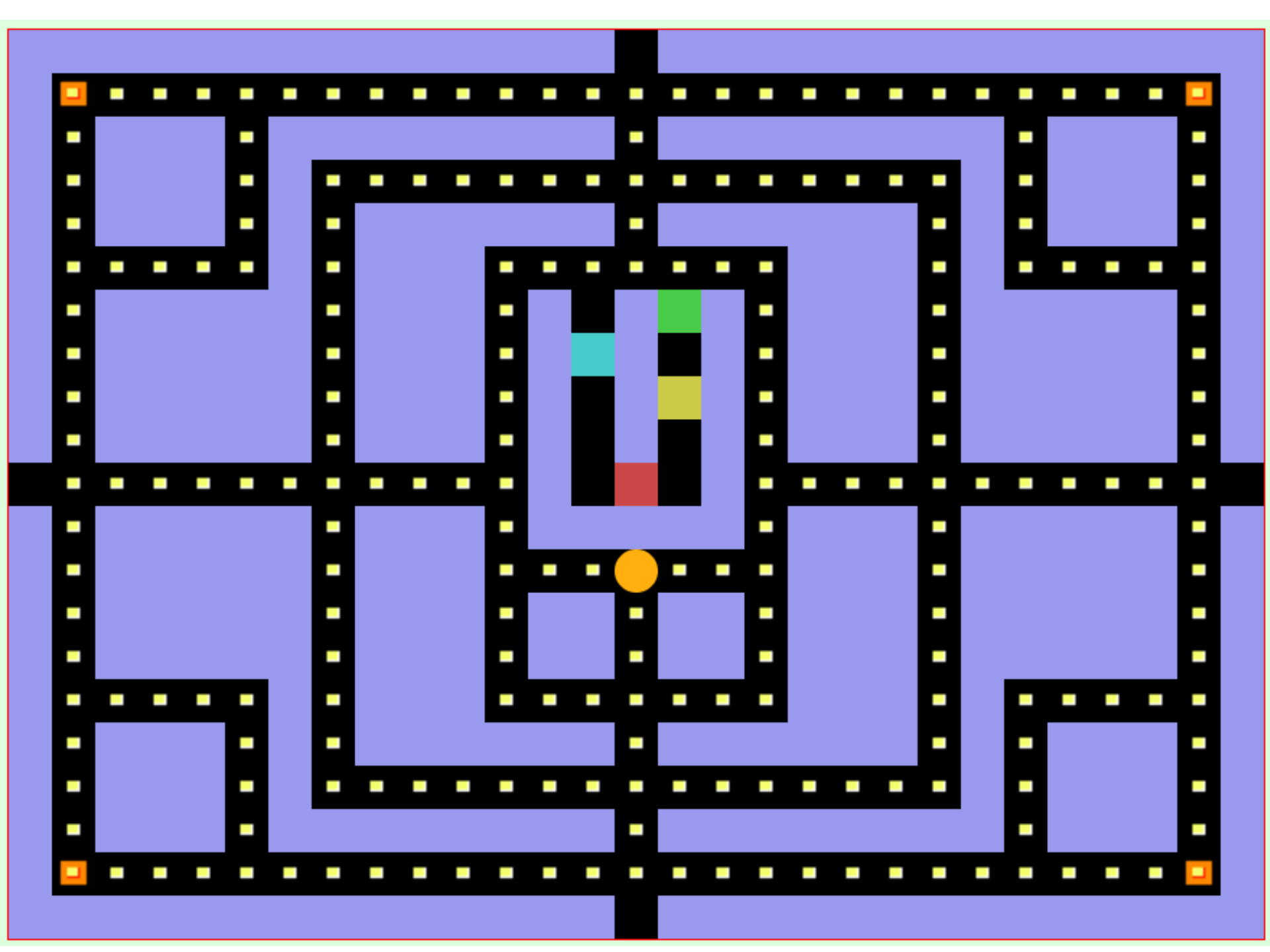
MOVIMENTAÇÃO FANTASMA



Ministério
da Educação

Movimentação Fantasma

- Para que o fantasma se mova, precisamos definir regras que de alguma forma demonstrem coerência no seu movimento;
- Por exemplo:
 - Não mudar de direção a todo o segundo;
 - Ser capaz de eventualmente trocar de direção;
 - Não atravessar as paredes;
 - Entre outros.
- Antes de mover, o fantasma precisa verificar as possibilidades para onde pode se mover.



Ghost.js (1/4)

```
//Atributo dinâmico
this.listaDirecoes = new Array();

this.checarDirecoes = function() {
    //Limpar o array de possíveis direções
    this.listaDirecoes.length = 0;
    if (this.direcaoAtual != Direcao.naoDefinida) {
        this.listaDirecoes.push(this.direcaoAtual);
    }
    if (this.direcaoAtual != Direcao.cima && this.direcaoAtual !=
        Direcao.baixo) {
        this.listaDirecoes.push(Direcao.cima);
        this.listaDirecoes.push(Direcao.baixo);
    }
    if (this.direcaoAtual != Direcao.esquerda && this.direcaoAtual !=
        Direcao.direita) {
        this.listaDirecoes.push(Direcao.esquerda);
        this.listaDirecoes.push(Direcao.direita);
    }
}
```

Continua

Ghost.js (2/4)

```
var i = 0;
while (i < this.listaDirecoes.length) {
  var remover = false;
  switch (this.listaDirecoes[i]) {
    case Direcao.cima:
      if (this.y <= 1) {
        remover = true;
      } else {
        if (Cenario.mapa[this.y - 1][this.x] ==
                                                    Cenario.parede) {
          remover = true;
        }
      }
    }
  break;
}
```

Continua

Ghost.js (3/4)

```
case Direcao.baixo:
    if (this.y >= ny - 2) {
        remover = true;
    } else {
        if (Cenario.mapa[this.y + 1][this.x] == Cenario.parede) {
            remover = true;
        }
    }
    break;
case Direcao.esquerda:
    if (this.x <= 1) {
        remover = true;
    } else {
        if (Cenario.mapa[this.y][this.x - 1] == Cenario.parede) {
            remover = true;
        }
    }
    break;
```

Continua

Ghost.js (4/4)

```
case Direcao.direita:
    if (this.x >= nx - 2) {
        remover = true;
    } else {
        if (Cenario.mapa[this.y][this.x + 1] == Cenario.parede) {
            remover = true;
        }
    }
    break;
} //Fim do switch
if (remover) {
    this.listaDirecoes.splice(i, 1);
} else {
    i++;
}
} //Fim do WHILE
} //Fim da função checarDirecoes()
```

Fim

Mover

- Depois de criar uma lista de possíveis movimentos, o Fantasma deve ter a possibilidade de optar em seguir na mesma direção, mas também de mudar o seu caminho;
- Em nossa implementação, daremos 50% de chance de ele seguir para o mesmo sentido, os demais 50% serão divididos entre os demais possíveis caminhos.

Ghost.js (1/2)

```
this.mover = function() {  
    this.checarDirecoes();  
    var movimento = Direcao.naoDefinida;  
    var aleatorio = Math.random();  
    //Se o primeiro for sorteado ou a lista tiver apenas 1 opção  
    if (aleatorio < Ghost.chanceMovIgual || this.listaDirecoes.length  
                                              == 1) {  
        movimento = this.listaDirecoes[0];  
    } else {  
        chance = (1 - Ghost.chanceMovIgual) /  
                (this.listaDirecoes.length - 1);  
        for (ca = 1; ca < this.listaDirecoes.length; ca++) {  
            if (aleatorio < Ghost.chanceMovIgual + (ca * chance)) {  
                movimento = this.listaDirecoes[ca];  
                break;  
            }  
        }  
    }  
}
```

Continua

Ghost.js (2/2)

```
this.direcaoAtual = movimento;  
switch(movimento) {  
  case Direcao.cima:  
    this.y--;  
    break;  
  case Direcao.baixo:  
    this.y++;  
    break;  
  case Direcao.esquerda:  
    this.x--;  
    break;  
  case Direcao.direita:  
    this.x++;  
    break;  
  default:  
    break;  
}  
} //Fim função mover()
```

```
//Elementos estáticos da classe Ghost  
Ghost.chanceMovIgual = 0.50;
```

Fim



INSTITUTO FEDERAL
SANTA CATARINA

pacman.js

```
function moverGhosts() {  
    for (i = 0; i < ghosts.length; i++) {  
        ghosts[i].mover();  
    }  
}
```



Ministério
da Educação

Pac-man

- Colocar temporariamente os comandos dentro, no fim da função onKD():
 - moverGhosts();
 - moverPacman();
 - desenharTudo();
- Teste sua página!





INSTITUTO FEDERAL
SANTA CATARINA



Pac-man

CONTROLE GLOBAL DO MOVIMENTO



Ministério
da Educação

Movendo com Tempo

- Todo tipo de movimento tem uma velocidade;
- Como determinamos a velocidade de algum objeto?
 - Medida Espacial / Tempo!
 - KM/h
 - m/s
 - ...



Controlando o Tempo

- Como já definimos, a unidade de espaço de cada movimento dos fantasmas e do Pac-man será por posição;
- Agora precisamos determinar o *intervalo de tempo* que nosso jogo irá usar para fazer cada movimento dos elementos;
- Como nosso jogo gira em torno do Pac-man e dos Fantasmas (e cada um terá uma velocidade diferente), estes tempos serão como guias para todo o jogo.



Controlando o Tempo

- Função JavaScript:
 - `relogio = setInterval("NomeFuncao()", intervalo);`
 - **relogio** é uma referência ao timer/clock que foi criado;
 - **NomeFuncao()** é a função que será executada a cada intervalo;
 - **intervalo** é um número inteiro representando a quantidade em milissegundos de intervalo entre uma execução e outra da função `NomeFuncao()`.
 - `clearInterval(relogio);`
 - Para o relógio de repetição;



configPacman.js

- Crie a variável:
 - **var** intervalo = 200;
- Esta variável irá definir a quantidade de milissegundos entre cada movimento do Pac-man;
- O intervalo entre os movimentos dos fantasmas será um pouco maior, para deixá-los mais lentos.

pacman.js

- Crie as variáveis:
 - **var** relógio = **null**;
 - **var** relógioGhosts = **null**;
- Elas serão responsáveis por armazenar a referência aos objetos que farão a temporização de movimento do Pac-man e dos fantasmas.

pacman.js

```
function pausar() {  
  if (relogio != null) {  
    clearInterval(relogio);  
    clearInterval(relogioGhosts);  
    relogio = null;  
    relogioGhosts = null;  
    btPausa.innerHTML = "Continuar";  
  } else {  
    relogio = setInterval("atualizaPacman()", intervalo);  
    relogioGhosts = setInterval("atualizaGhosts()",  
                                Math.round(intervalo * 1.2));  
    btPausa.innerHTML = "Pausar";  
  }  
}
```

Cada Fantasma é
20% mais lento
que Pac-man

pacman.js

```
function atualizaGhosts() {  
    moverGhosts();  
    desenharTudo();  
}
```

```
function atualizaPacman() {  
    moverPacman();  
    desenharTudo();  
}
```

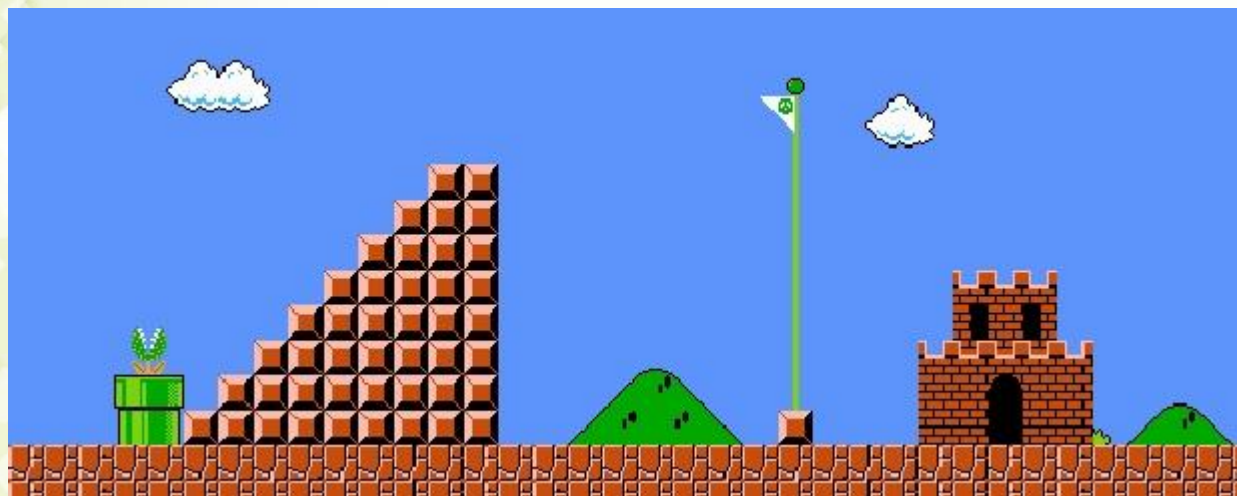
Pac-man

- Se você ainda não retirou os seguintes comandos de dentro da função onKD(), **retire**:
 - moverGhosts();
 - moverPacman();
 - desenharTudo();
- Teste sua página!



Alterações Restantes

- O que falta alterar?
 - Comer os pontos?
 - Comer os pontos de poder?
 - Assustar os fantasmas!
 - Devorar os fantasmas!
 - Fim de jogo?
 - Sons?



pacman.js

```
function atualizaGhosts() {  
    moverGhosts();  
    if (verificaColisoas()) {  
        gameOver();  
    }  
    desenharTudo();  
}  
  
function atualizaPacman() {  
    moverPacman();  
    if (verificaColisoas()) {  
        gameOver();  
    }  
    desenharTudo();  
}
```

pacman.js (1/2)

```
//Retorna verdadeiro para o caso de Game Over
function verificaColisoes() {
    //Comer ponto?
    if (Cenario.mapa[py][px] == Cenario.ponto) {
        Cenario.mapa[py][px] = Cenario.vazio;
        //Ponto do poder?
    } else if (Cenario.mapa[py][px] == Cenario.poder) {
        Cenario.mapa[py][px] = Cenario.vazio;
        for (i = 0; i < ghosts.length; i++) {
            ghosts[i].assustar();
        }
    } //Fim do else if
}
```

pacman.js (2/2)

```
//Colisão com fantasmas
```

```
for (i = 0; i < ghosts.length; i++) {  
    if (px == ghosts[i].x && py == ghosts[i].y) {  
        if (ghosts[i].assustado == 0) {  
            return true;  
        } else {  
            ghosts[i].devorado();  
        }  
    }  
}  
  
return false;  
} //Fim da função
```



INSTITUTO FEDERAL
SANTA CATARINA

pacman.js



GAME OVER

```
function gameOver() {  
    pausar();  
    btPausa.disabled = true;  
    btPausa.innerHTML = "Game Over!";  
}
```



Ministério
da Educação



INSTITUTO FEDERAL
SANTA CATARINA

Ghost.js

```
//Número de movimentos restantes como assustado
this.assustado = 0;
this.assustar = function () {
    this.assustado = 30;
    switch(this.direcaoAtual) {
        case Direcao.cima:
            this.direcaoAtual = Direcao.baixo;
            break;
        case Direcao.baixo:
            this.direcaoAtual = Direcao.cima;
            break;
        case Direcao.esquerda:
            this.direcaoAtual = Direcao.direita;
            break;
        case Direcao.direita:
            this.direcaoAtual = Direcao.esquerda;
            break;
    }
}
```



Ministério
da Educação



INSTITUTO FEDERAL
SANTA CATARINA

Ghost.js

```
//Quando o Fantasma for devorado
this.devorado = function () {
    this.assustado = 0;
    this.x = this.xi;
    this.y = this.yi;
}

//Adicionar à função mover()
this.mover = function() {
    if (this.assustado > 0) {
        this.assustado--;
    }
    this.checarDirecoes();
    ...
}
```



Ministério
da Educação



INSTITUTO FEDERAL
SANTA CATARINA

Ghost.js

```
//Alterar a função para desenhar fantasma assustado
this.desenhar = function(ct) {
    if (this.assustado == 0) {
        ct.fillStyle = this.cor;
    } else {
        ct.fillStyle = "rgba(255, 255, 255, 0.8)";
    }
    ct.fillRect(this.x * largura, this.y * largura,
                largura, largura);
}
```



Ministério
da Educação



INSTITUTO FEDERAL
SANTA CATARINA

Pac-man

- Teste sua página!



Ministério
da Educação



INSTITUTO FEDERAL
SANTA CATARINA

Pac-man

ESTÍMULOS SONOROS



Ministério
da Educação

Estímulos Sonoros

- Conforme comentado anteriormente, quanto mais estimularmos, de forma positiva, os sentidos dos jogadores, maior a probabilidade dele se sentir como parte do jogo;
- Para isto, iremos adicionar alguns pequenos sons associados a eventos como colisões;
- Baixe os arquivos e salve na subpasta **snd**:
 - wakka.mp_;
 - wakka.ogg;

<audio> e <source>

- HTML 5!
- MIME Types:
 - MP3 – audio/mpeg
 - Ogg – audio/ogg
 - Wav – audio/wav
- Suporte:
 - Ps.: Múltiplos <source> fornecem redundância!

Browser	MP3	Wav	Ogg
IE 9+	Sim	Não	Não
Chrome 6+	Sim	Sim	Sim
Firefox 3.6+	Não	Sim	Sim
Safari 5+	Sim	Sim	Não
Opera 10+	Não	Sim	Sim



INSTITUTO FEDERAL
SANTA CATARINA

index.html

...

```
<audio controls id="waka">  
  <source src="snd/waka.mp_" type="audio/mpeg">  
  <source src="snd/waka.ogg" type="audio/ogg">  
</audio>  
<script src="pacman.js"></script>
```



Ministério
da Educação

pacman.js

```
//Recuperando referência dos objetos no documento  
var canvas = document.getElementById("tela");  
var ctx = canvas.getContext("2d");  
var btPausa = document.getElementById("btPausa");  
var btNovo = document.getElementById("btNovo");  
var audioWaka = document.getElementById("waka");
```

Prática: Insira os comandos de reprodução nos locais apropriados:

Ex.:

```
audioWaka.currentTime = 0;  
audioWaka.play();
```



Trabalho

1. Customize cores e outras configurações do arquivo de estilo;
2. Customize cores, tamanhos e disposição dos objetos do jogo (dentro do *Javascript*). Utilize gradientes e/ou imagens;
3. Complete o HTML informando o nome da disciplina, o nome do instituto e o seu nome, dispondo os elementos com layouts CSS;
4. Existe um **bug**: quando o jogo não está em execução, teclas pressionadas são processadas ao retomar o jogo. Corrija este bug.
5. Bug: Quando o jogo está em execução, ao pressionar o botão “Novo Jogo” o jogo reinicia mas não para novo início.
6. Existe um problema de rolagem quando são utilizadas as setas para controlar o jogo e a página ocupa um espaço maior que a tela. Resolva este problema.
7. Crie um placar com pontuação;
8. Crie uma indicação visual dentro do Canvas de **fim de jogo**;
9. Adicione novos sons a eventos diferentes no jogo;
10. Adicione teclas de atalho para “Pausa” e para “Novo Jogo”;
11. Ao evoluir no jogo, crie novos desafios para o jogador;
12. Adicione outros elementos a seu critério (P/E);
13. Entregue os arquivos por e-mail ao Professor junto com uma **descrição**.

S

P/E