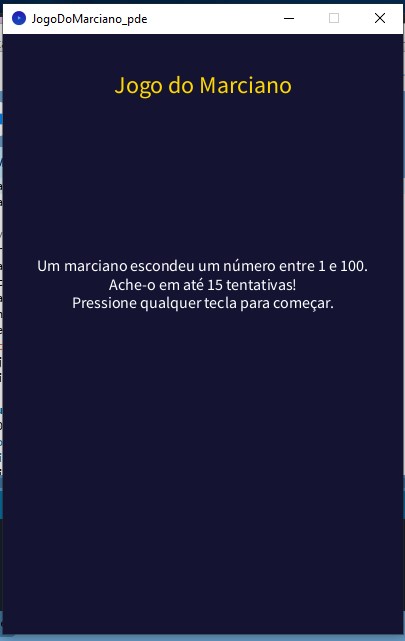
**DG – Relatório de entrega da 1° avaliação – Trabalhos 1, 2 e 3 (Unidade 1)**

**Nome do Game Studio:** Calango Banguela   
  
**Link do Git Hub:** https://github.com/AdamCaldas/jogos-processing  
  
**Integrantes:**   
- Adam Caldas Lopes - 01640350   
- Antônio Henrique Leitão Barros - 01647043   
- Emanuel Ernesto de Andrade Rego - 01614951  
- Felipe Farias Galvao - 01608490   
- Guilherme de Luna Barros – 01615925  
- José Carlos Moura Silva - 01530501  
- Thiago Oliveira Lima Pessoa - 01089643   
- Wesley Ruan de Lima Silva - 01555915

**1° Jogo do Marciano.**  
  
Prints das telas:  
  
Tela Inicial:



Segunda Tela:



Terceira Tela:   
  
  
  
Código do jogo:  
  
import java.util.ArrayList;



import java.util.Collections;

// Variáveis globais

int numeroSecreto;

int tentativas;

int limiteTentativas = 15;

boolean acertou = false;

String entrada = "";

String mensagem = "";

ArrayList<Integer> recordes = new ArrayList<Integer>();

boolean jogoAtivo = true;

boolean introducaoAtiva = true;

void setup() {

size(400, 600);

background(0);

textAlign(CENTER, CENTER);

reiniciarJogo();

}

void draw() {

background(20, 20, 50);

if (introducaoAtiva) {

// Tela de introdução

fill(255, 215, 0);

textSize(24);

text("Jogo do Marciano", width/2, 50);

fill(255);

textSize(16);

text("Um marciano escondeu um número entre 1 e 100.\n" +

"Ache-o em até " + limiteTentativas + " tentativas!\n" +

"Pressione qualquer tecla para começar.", width/2, height/2 - 50);

} else {

// Tela do jogo

fill(255);

textSize(20);

text("Tentativas restantes: " + (limiteTentativas - tentativas), width/2, 50);

textSize(18);

text("Digite sua tentativa (1-100):", width/2, height/2 - 80);

fill(200, 255, 200); // Verde claro

text(entrada, width/2, height/2 - 50);

fill(255, 165, 0); // Laranja

textSize(20);

text(mensagem, width/2, height/2);

// Recordes

if (!recordes.isEmpty()) {

fill(255);

textSize(16);

text("Melhores Jogadas:", width/2, height - 120);

Collections.sort(recordes);

for (int i = 0; i < min(3, recordes.size()); i++) {

fill(173, 216, 230); // Azul claro

text((i + 1) + "º: " + recordes.get(i) + " tentativas", width/2, height - 100 + i \* 20);

}

}

// Instruções no rodapé

fill(255);

textSize(14);

if (jogoAtivo) {

text("Pressione 'Z' para voltar ao início", width/2, height - 20);

} else {

text("Pressione 'R' para jogar novamente", width/2, height - 20);

}

}

}

void keyPressed() {

if (introducaoAtiva) {

// Qualquer tecla sai da introdução

introducaoAtiva = false;

jogoAtivo = true;

} else if (jogoAtivo) {

// Entrada de números

if (key >= '0' && key <= '9') {

entrada += key;

}

// Apagar com Backspace

else if (key == BACKSPACE && entrada.length() > 0) {

entrada = entrada.substring(0, entrada.length() - 1);

}

// Processar tentativa com Enter

else if (key == ENTER || key == RETURN) {

if (entrada.length() > 0) {

int tentativa = int(entrada);

tentativas++;

if (tentativa == numeroSecreto) {

mensagem = "Parabéns! Acertou em " + tentativas + " tentativas!";

recordes.add(tentativas);

acertou = true;

jogoAtivo = false;

} else if (tentativa < numeroSecreto) {

mensagem = "O número é MAIOR!";

} else {

mensagem = "O número é MENOR!";

}

if (tentativas >= limiteTentativas && !acertou) {

mensagem = "Fim! O número era " + numeroSecreto;

jogoAtivo = false;

}

entrada = "";

}

}

// Voltar à introdução com 'Z'

else if (key == 'z' || key == 'Z') {

introducaoAtiva = true;

jogoAtivo = false;

reiniciarJogo();

}

} else if (key == 'r' || key == 'R') {

// Reinicia o jogo diretamente

reiniciarJogo();

introducaoAtiva = false;

jogoAtivo = true;

}

}

void reiniciarJogo() {

numeroSecreto = int(random(1, 101));

tentativas = 0;

acertou = false;

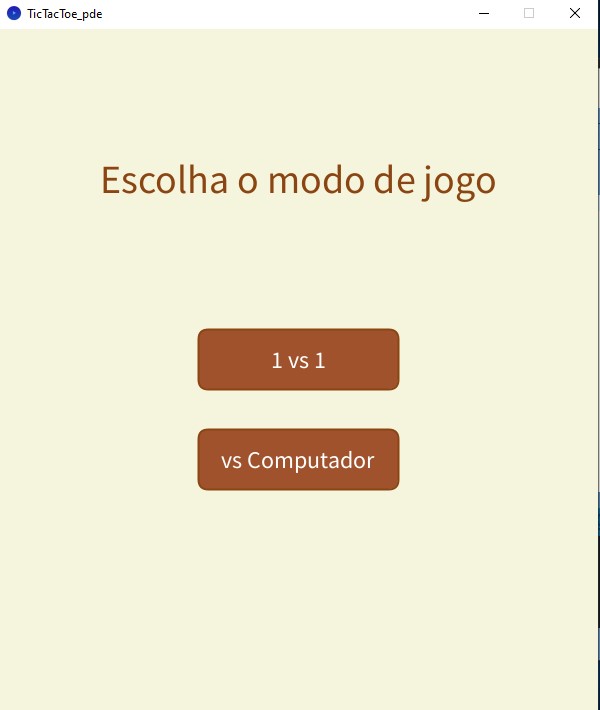
entrada = "";

mensagem = "";

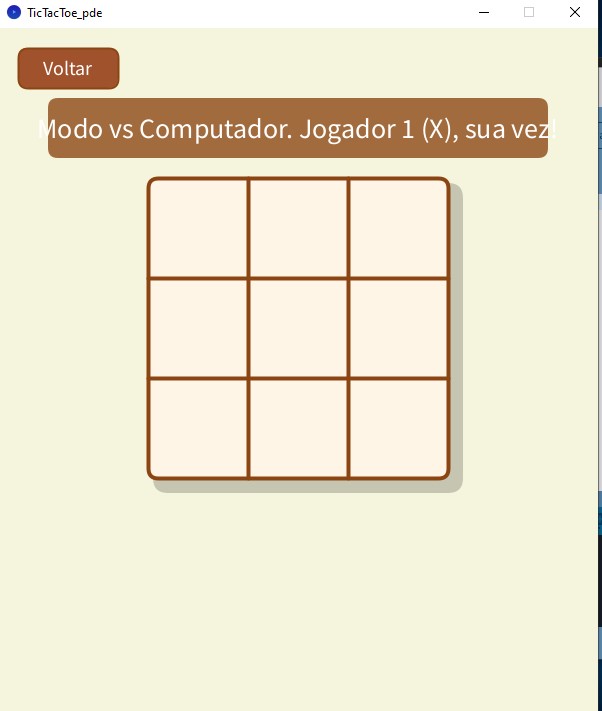
}  
  
  
**2° Jogo da Velha (TicTacToe).**  
  
Prints das telas:  
  
Tela Inicial:



Segunda Tela:



Terceira Tela:



Quarta Tela:



Código do jogo:

char[][] board = new char[3][3];

boolean playerVsPlayer = false;

int currentPlayer = 1;

boolean gameOver = false;

String message = "Jogador 1 (X), sua vez!";

int screen = 0;

void setup() {

size(600, 700);

resetGame();

textAlign(CENTER, CENTER);

}

void resetGame() {

for (int i = 0; i < 3; i++) {

for (int j = 0; j < 3; j++) {

board[i][j] = ' ';

}

}

currentPlayer = 1;

gameOver = false;

message = "Jogador 1 (X), sua vez!";

}

void draw() {

if (screen == 0) {

drawWelcomeScreen();

} else if (screen == 1) {

drawMenuScreen();

} else if (screen == 2) {

drawGameScreen();

}

}

void drawWelcomeScreen() {

background(245, 245, 220);

fill(139, 69, 19);

textSize(50);

text("Bem-vindo ao", width/2, height/2 - 50);

text("Jogo da Velha", width/2, height/2 + 10);

fill(160, 82, 45);

textSize(30);

text("Aperte Enter para começar", width/2, height/2 + 100);

}

void drawMenuScreen() {

background(245, 245, 220);

fill(139, 69, 19);

textSize(40);

text("Escolha o modo de jogo", width/2, 150);

drawOptionButton(200, 300, 200, 60, "1 vs 1");

drawOptionButton(200, 400, 200, 60, "vs Computador");

}

void drawGameScreen() {

background(245, 245, 220);

noStroke();

fill(0, 0, 0, 50);

rect(155, 155, 310, 310, 15);

strokeWeight(4);

stroke(139, 69, 19);

fill(255, 245, 230);

rect(150, 150, 300, 300, 10);

line(250, 150, 250, 450);

line(350, 150, 350, 450);

line(150, 250, 450, 250);

line(150, 350, 450, 350);

for (int i = 0; i < 3; i++) {

for (int j = 0; j < 3; j++) {

float x = 200 + j \* 100;

float y = 200 + i \* 100;

if (board[i][j] == 'X') {

drawX(x, y);

} else if (board[i][j] == 'O') {

drawO(x, y);

}

}

}

fill(139, 69, 19, 200);

noStroke();

rect(50, 70, 500, 60, 10);

fill(255);

textSize(28);

text(message, width/2, 100);

drawBackButton(20, 20, 100, 40);

}

void drawOptionButton(int x, int y, int w, int h, String label) {

boolean hover = mouseX > x && mouseX < x + w && mouseY > y && mouseY < y + h;

if (hover) {

fill(210, 180, 140);

} else {

fill(160, 82, 45);

}

stroke(139, 69, 19);

strokeWeight(2);

rect(x, y, w, h, 10);

fill(255);

textSize(24);

text(label, x + w/2, y + h/2);

}

void drawBackButton(int x, int y, int w, int h) {

boolean hover = mouseX > x && mouseX < x + w && mouseY > y && mouseY < y + h;

if (hover) {

fill(210, 180, 140);

} else {

fill(160, 82, 45);

}

stroke(139, 69, 19);

strokeWeight(2);

rect(x, y, w, h, 10);

fill(255);

textSize(20);

text("Voltar", x + w/2, y + h/2);

}

void drawX(float x, float y) {

stroke(220, 20, 60);

strokeWeight(10);

line(x - 35, y - 35, x + 35, y + 35);

line(x + 35, y - 35, x - 35, y + 35);

stroke(255, 100, 120, 150);

strokeWeight(6);

line(x - 35, y - 35, x + 35, y + 35);

line(x + 35, y - 35, x - 35, y + 35);

}

void drawO(float x, float y) {

noFill();

stroke(70, 130, 180);

strokeWeight(10);

ellipse(x, y, 70, 70);

stroke(135, 206, 235, 150);

strokeWeight(6);

ellipse(x, y, 70, 70);

}

void keyPressed() {

if (screen == 0 && key == ENTER) {

screen = 1;

}

}

void mousePressed() {

if (screen == 1) {

if (mouseY > 300 && mouseY < 360 && mouseX > 200 && mouseX < 400) {

playerVsPlayer = true;

screen = 2;

resetGame();

message = "Modo vs Jogador. Jogador 1 (X), sua vez!";

} else if (mouseY > 400 && mouseY < 460 && mouseX > 200 && mouseX < 400) {

playerVsPlayer = false;

screen = 2;

resetGame();

message = "Modo vs Computador. Jogador 1 (X), sua vez!";

}

} else if (screen == 2) {

if (mouseX > 20 && mouseX < 120 && mouseY > 20 && mouseY < 60) {

screen = 1;

resetGame();

return;

}

if (gameOver) {

resetGame();

return;

}

if (mouseX > 150 && mouseX < 450 && mouseY > 150 && mouseY < 450) {

int row = (mouseY - 150) / 100;

int col = (mouseX - 150) / 100;

if (board[row][col] == ' ') {

if (currentPlayer == 1) {

board[row][col] = 'X';

message = "Jogador 2 (O), sua vez!";

} else {

board[row][col] = 'O';

message = "Jogador 1 (X), sua vez!";

}

if (checkWin()) {

message = "Jogador " + currentPlayer + " venceu!";

gameOver = true;

} else if (isBoardFull()) {

message = "Empate!";

gameOver = true;

} else {

currentPlayer = (currentPlayer == 1) ? 2 : 1;

if (!playerVsPlayer && currentPlayer == 2) {

computerMove();

}

}

}

}

}

}

void computerMove() {

if (gameOver) return;

ArrayList<int[]> emptyCells = new ArrayList<int[]>();

for (int i = 0; i < 3; i++) {

for (int j = 0; j < 3; j++) {

if (board[i][j] == ' ') {

emptyCells.add(new int[]{i, j});

}

}

}

if (!emptyCells.isEmpty()) {

int[] move = emptyCells.get((int)random(emptyCells.size()));

board[move[0]][move[1]] = 'O';

if (checkWin()) {

message = "Computador venceu!";

gameOver = true;

} else if (isBoardFull()) {

message = "Empate!";

gameOver = true;

} else {

currentPlayer = 1;

message = "Jogador 1 (X), sua vez!";

}

}

}

boolean checkWin() {

for (int i = 0; i < 3; i++) {

if (board[i][0] != ' ' && board[i][0] == board[i][1] && board[i][1] == board[i][2]) return true;

if (board[0][i] != ' ' && board[0][i] == board[1][i] && board[1][i] == board[2][i]) return true;

}

if (board[0][0] != ' ' && board[0][0] == board[1][1] && board[1][1] == board[2][2]) return true;

if (board[0][2] != ' ' && board[0][2] == board[1][1] && board[1][1] == board[2][0]) return true;

return false;

}

boolean isBoardFull() {

for (int i = 0; i < 3; i++) {

for (int j = 0; j < 3; j++) {

if (board[i][j] == ' ') return false;

}

}

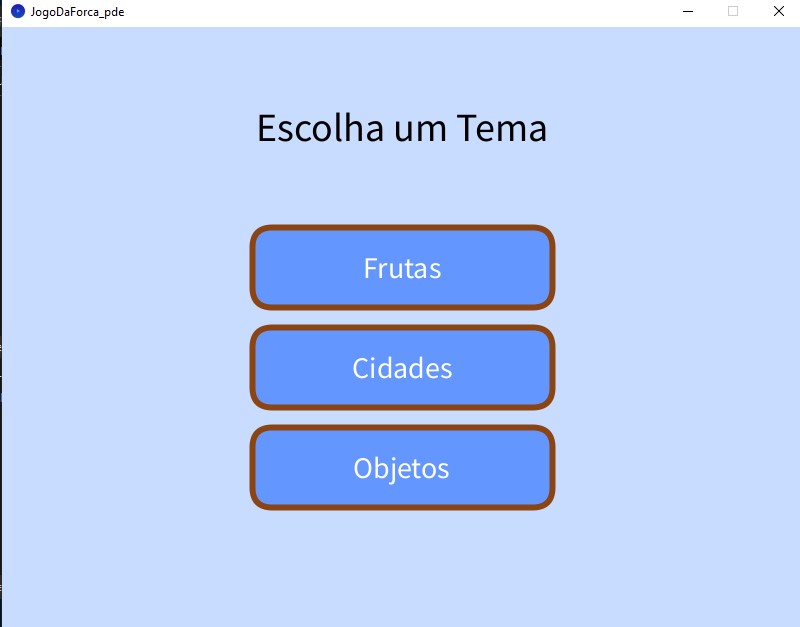
return true;

}

**3° Jogo da Forca.**  
  
Prints das telas:  
  
Tela Inicial:

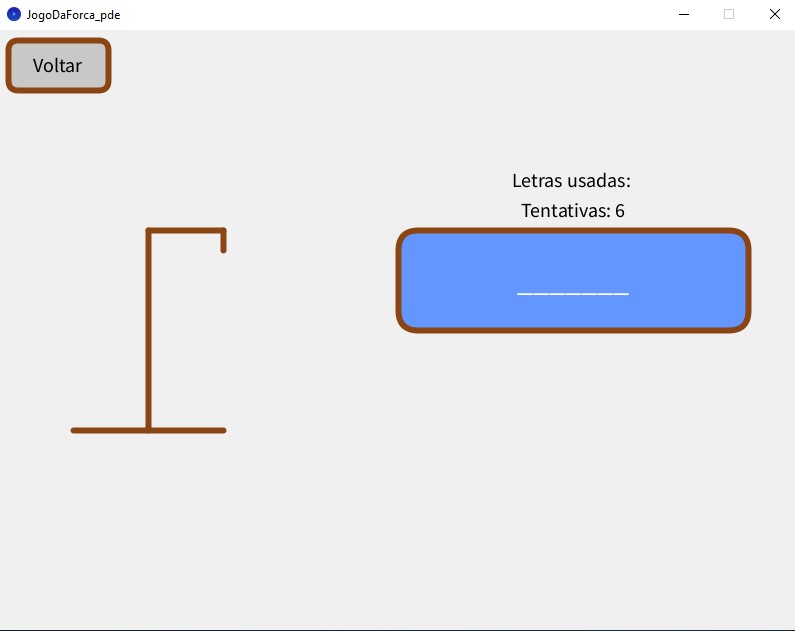


Segunda Tela:



Terceira Tela:

Código do jogo:  
  
// Declaração das variáveis globais



String[] frutas = {"maça", "banana", "laranja", "uva", "manga"};

String[] cidades = {"recife", "são paulo", "rio", "manaus", "belém"};

String[] objetos = {"mesa", "cadeira", "lápis", "caneta", "livro"};

String[] palavras;

String palavraSecreta;

String palavraAtual;

int tentativasRestantes = 6;

boolean jogoAcabou = false;

String mensagem = "";

char[] letrasUsadas = new char[26];

int letrasUsadasCount = 0;

int tela = 0;

void setup() {

size(800, 600);

textAlign(CENTER, CENTER);

}

void draw() {

if (tela == 0) {

desenharTelaInicial();

} else if (tela == 1) {

desenharMenu();

} else if (tela == 2) {

desenharJogo();

}

}

void desenharTelaInicial() {

background(50, 150, 200);

fill(255);

textSize(40);

text("Jogo da Forca", width/2, height/3);

fill(0, 200, 0);

rect(width/2 - 100, height/2, 200, 80, 20);

fill(255);

textSize(30);

text("Iniciar", width/2, height/2 + 40);

stroke(255);

strokeWeight(2);

noFill();

ellipse(width/2, height/3, 300, 100);

}

void desenharMenu() {

background(200, 220, 255);

fill(0);

textSize(40);

text("Escolha um Tema", width/2, 100);

desenharBotao("Frutas", width/2 - 150, 200, 300, 80);

desenharBotao("Cidades", width/2 - 150, 300, 300, 80);

desenharBotao("Objetos", width/2 - 150, 400, 300, 80);

}

void desenharBotao(String texto, float x, float y, float w, float h) {

fill(100, 150, 255);

rect(x, y, w, h, 20);

fill(255);

textSize(30);

text(texto, x + w/2, y + h/2);

}

void desenharJogo() {

background(240);

// Botão de voltar ao menu

fill(200);

rect(10, 10, 100, 50, 10);

fill(0);

textSize(20);

text("Voltar", 60, 35);

stroke(0);

strokeWeight(2);

desenharForca(6 - tentativasRestantes);

// Área à direita para a palavra (fundo azul)

fill(100, 150, 255);

rect(400, 200, 350, 100, 20);

fill(255);

textSize(32);

text(palavraAtual, 575, 250);

String usadas = "Letras usadas: ";

for (int i = 0; i < letrasUsadasCount; i++) {

usadas += letrasUsadas[i] + " ";

}

fill(0);

textSize(20);

text(usadas, 575, 150);

text("Tentativas: " + tentativasRestantes, 575, 180);

if (jogoAcabou) {

fill(0, 0, 255);

textSize(28);

if (tentativasRestantes > 0) {

text("Você venceu! Palavra: " + palavraSecreta, 575, 100);

} else {

text("Você perdeu! Palavra: " + palavraSecreta, 575, 100);

}

text("Pressione 'R' para recomeçar", 575, 500);

}

}

void mousePressed() {

if (tela == 0) {

if (mouseX > width/2 - 100 && mouseX < width/2 + 100 &&

mouseY > height/2 && mouseY < height/2 + 80) {

tela = 1;

}

} else if (tela == 1) {

if (mouseX > width/2 - 150 && mouseX < width/2 + 150) {

if (mouseY > 200 && mouseY < 280) {

palavras = frutas;

iniciarJogo();

} else if (mouseY > 300 && mouseY < 380) {

palavras = cidades;

iniciarJogo();

} else if (mouseY > 400 && mouseY < 480) {

palavras = objetos;

iniciarJogo();

}

}

} else if (tela == 2) {

if (mouseX > 10 && mouseX < 110 && mouseY > 10 && mouseY < 60) {

tela = 1;

}

}

}

void keyPressed() {

if (tela == 2 && !jogoAcabou && key >= 'a' && key <= 'z') {

char letra = key;

boolean jaUsada = false;

for (int i = 0; i < letrasUsadasCount; i++) {

if (letrasUsadas[i] == letra) {

jaUsada = true;

break;

}

}

if (!jaUsada) {

letrasUsadas[letrasUsadasCount] = letra;

letrasUsadasCount++;

boolean acertou = false;

String novaPalavraAtual = "";

for (int i = 0; i < palavraSecreta.length(); i++) {

if (palavraSecreta.charAt(i) == letra) {

novaPalavraAtual += letra;

acertou = true;

} else {

novaPalavraAtual += palavraAtual.charAt(i);

}

}

palavraAtual = novaPalavraAtual;

if (!acertou) {

tentativasRestantes--;

}

if (tentativasRestantes <= 0) {

jogoAcabou = true;

} else if (!palavraAtual.contains("\_")) {

jogoAcabou = true;

}

}

}

if (tela == 2 && jogoAcabou && key == 'r') {

iniciarJogo();

}

}

void iniciarJogo() {

palavraSecreta = palavras[int(random(palavras.length))];

palavraAtual = "";

for (int i = 0; i < palavraSecreta.length(); i++) {

palavraAtual += "\_";

}

tentativasRestantes = 6;

jogoAcabou = false;

mensagem = "";

letrasUsadasCount = 0;

tela = 2;

}

void desenharForca(int erros) {

stroke(139, 69, 19);

strokeWeight(6);

float xBase = 150;

float yBase = 300;

line(xBase - 75, yBase + 100, xBase + 75, yBase + 100);

line(xBase, yBase + 100, xBase, yBase - 100);

line(xBase, yBase - 100, xBase + 75, yBase - 100);

line(xBase + 75, yBase - 100, xBase + 75, yBase - 80);

//corpo

if (erros >= 1) {

noFill();

ellipse(xBase + 75, yBase - 50, 80, 80);

}

if (erros >= 2) {

line(xBase + 75, yBase - 10, xBase + 75, yBase + 50);

}

if (erros >= 3) {

line(xBase + 75, yBase, xBase + 35, yBase - 40);

}

if (erros >= 4) {

line(xBase + 75, yBase, xBase + 115, yBase - 40);

}

if (erros >= 5) {

line(xBase + 75, yBase + 50, xBase + 35, yBase + 90);

}

if (erros >= 6) {

line(xBase + 75, yBase + 50, xBase + 115, yBase + 90);

}

}