

**IFSP- INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E
TECNOLOGIA SÃO PAULO.**

TÉCNICO INTEGRADO EM INFORMÁTICA

EDUARDA BOMFIM DA CONCEIÇÃO Nº 06

FERNANDA APARECIDA FIGUEIREDO DA SILVA Nº 08

GIOVANA PAZ PEDROZO Nº 12

JULIA GOMES NASCIMENTO Nº 16

VITÓRIA DA SILVA MOÇO Nº 34

BIO-LÓGICAS

SÃO PAULO

2018

IFSP- INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA SÃO PAULO.

TÉCNICO INTEGRADO EM INFORMÁTICA

EDUARDA BOMFIM DA CONCEIÇÃO Nº 06

FERNANDA APARECIDA FIGUEIREDO DA SILVA Nº 08

GIOVANA PAZ PEDROZO Nº 12

JULIA GOMES NASCIMENTO Nº 16

VITÓRIA DA SILVA MOÇO Nº 34

BIO- LÓGICAS

Trabalho de lógica solicitado pelas professoras Claudia Miyuki Werhmuller, Livia Grulli e Beatriz Torrano, como exigência para obtenção de aprovação para o 4º bimestre do ano de 2018.

SÃO PAULO

2018

RESUMO:

O presente trabalho descreve todo o processo de realização do projeto que agrega as disciplinas de Biologia e Lógica de Programação 1: Bio-Lógicas.

De forma detalhada, todos os processos de desenvolvimento do projeto estão presentes aqui, de maneira que os aprendizados e pesquisas realizados durante o mesmo sejam evidenciados; possibilitando a compreensão de toda a banca de avaliação, de como foi possível desenvolver e finalizar tal projeto.

O jogo “Bio-Lógicas” é uma maneira lúdica e divertida de aprender sobre os conceitos de biomas mundiais e suas características, através do reconhecimento de padrões presentes em cada bioma de acordo com sua fauna, flora ou clima.

SUMÁRIO

1	INTRODUÇÃO	4
2	PARTE LÓGICA	5
3	PARTE DE BIOLOGIA.....	6
4	CONCLUSÃO	9
	FONTES BIBLIOGRÁFICAS E WEBGRÁFICAS	10
	BIBLIOGRÁFICAS:	10
	WEBGRÁFICAS:	10
	ANEXO A – CÓDIGO DO JOGO.....	12

1 INTRODUÇÃO

Em todas as fases do jogo “Bio-Lógicas”, é informado ao jogador um bioma diferente do anterior. A partir disso, são mostradas seis imagens; onde cinco correspondem a características do bioma dado, e uma, pode ser considerada intrusa naquele ambiente, ou seja, não corresponde àquele bioma, sendo em sua fauna e flora ou em seu clima.

Pelo fato de que seus níveis avançados exigem um pouco mais de conhecimento, Bio-Lógicas é indicado para maiores de 10 anos. Apesar disso, apresenta simplicidade em seu modo de se apresentar ao usuário; com tela de auxílio e bastante estímulo visual, torna-se fácil jogá-lo, não sendo impossível que crianças menores o joguem, apenas mais difícil.

2 PARTE LÓGICA

O jogo foi desenvolvido em linguagem C, utilizando a biblioteca gráfica “Allegro 4” no ambiente de desenvolvimento “Dev C++”, versão 4.9.9.2.

Para a criação da interface gráfica do projeto, foi utilizado imagens, no formato “BITMAP”, com a função “*load_bitmap*”; também foi utilizado o download de fontes no formato “PCX”, com a função “*load_font*”; e para complementar o jogo e deixá-lo mais dinâmico, foi utilizado sons de fundo e efeitos sonoros, no formato “WAV”, com a função “*load_sample*”. Para realizar a troca de tela, foi utilizada a função “*enum{}*”.

Apesar de contar com uma biblioteca gráfica, o projeto compila rapidamente, graças ao uso de ponteiros e a maior parte das variáveis serem do tipo “*int*” e locais. O código que conta com 1348 linhas é organizado pelo uso de funções para cada tela, instruindo dentro da *main()*, apenas a exibição de cada função correspondente a tela “chamada”.

Obs: todas estas funções utilizadas, são próprias da biblioteca “Allegro”.

Obs2: principais partes do código do jogo, vide em anexo, no final deste documento.

3 PARTE DE BIOLOGIA

Todo o jogo “Bio-Lógicas” baseia-se na matéria de biomas mundiais e suas características. Para que o jogo ganhasse um sentido didático, foi utilizada a definição de bioma, onde através deste, podemos perceber o padrão decorrente da associação consequentemente homogênea do equilíbrio natural entre a vegetação, o clima e os animais de cada bioma, permitindo assim, o reconhecimento do elemento não correspondente, através de suas diferenças que não se relacionam com as características daquele ambiente.

Biomas e elementos, correspondentes e não correspondentes de cada bioma, utilizados no jogo:

Amazônia

- Onça
- Arara vermelha
- Chuva
- Vegetação fechada
- Guaraná
- Canguru

Floresta temperada

- Panda vermelho
- Vegetação na primavera
- Lobo
- Veado
- Coníferas
- Foca

Deserto

- Camelo
- Cacto
- Víbora chifruda
- Arbustos

Savana

- Acácia
- Elefante
- Zebra
- Vegetação rasteira

- Solo seco

- Juazeiro

Taiga

- Alce

- Neve

- Raposa

- Esquilo

- Coníferas

- Zebra

Manguezal

- Caranguejo

- Mangue vermelho

- Mangue de botão

- Jacaré de papo amarelo

- Cobra-coral

- Cavalo

Pantanal

- Vitória-régia

- Arara-azul

- Girafa

- Golfinho

Pradarias

- Cavalo

- Avestruz

- Vegetação herbácea

- Búfalo

- Pica-pau

- Lobo

Tundra

- Vegetação gramínea

- Boi almiscarado

- Urso polar

- Rena

- Coruja das neves

- Avestruz

Chaparral

- Veado

- Azinheira

- Capivara

- Jenipapo

- Tuiuiú

- Urso

- Alecrim

- Coelho-do-chaparral

- Lagarto

- Tucano

4 CONCLUSÃO

Com este trabalho concluímos que, existem formas alternativas muito eficazes de aprendizado; como por exemplo, através deste jogo muito simples que ensina a entender os biomas de uma forma bem divertida. Com o grande uso de imagens, Bio-Lógicas pode estimular áreas do cérebro que facilitam a memorização e o aprendizado, tudo de uma maneira dinâmica e divertida, transformando a forma de aprender tradicional e exaustiva em algo que funcione subconscientemente, enquanto o jogador se diverte e testa suas habilidades.

Por isso, este jogo pode ser uma grande ferramenta de aprendizado, se usado de maneira didática e para expandir conhecimentos; ou apenas como forma de se divertir.

FONTES BIBLIOGRÁFICAS E WEBGRÁFICAS

BIBLIOGRÁFICAS:

FIGUEIRÓ, Adriano. Biogeografia: Dinâmicas E Transformações Da Natureza. São Paulo, Oficina de Textos, 2015.

HARBOUR, Jonathan. S.Game Programming All in One, Second Edition. Boston, MA: Thonson Course Technology, [2004].

RECURSOS NATURAIS E BIODIVERSIDADE, São Paulo, Editora Érica, 2014.

WEBGRÁFICAS:

ALLEGRO 4- TUTORIAL 14- SISTEMA DE TELAS. Disponível em:
<<https://www.youtube.com/watch?v=tR5rVSgde6A>>. Acesso em: 08 nov. 2018

ALLEGRO CC. Disponível em: <<https://www.allegro.cc/manual/4/api/fonts/>>. Acesso em: 27 out. 2018

ALLEGRO PARA INICIANTES. Disponível em: <<http://geocities.ws/weissengeist/allegro>>. Acesso em: 27 out. 2018

ALLEGRO.CC. Disponível em: <<https://www.allegro.cc/resource/Tools/Fonts/ttf2pcx>>. Acesso em: 18 out. 2018

ALLEGRO-CRIANDO JOGOS EM C++. Disponível em:
<https://www.clubedohardware.com.br/forums/topic/1189593-allegro-criando-jogos-em-c/>>. Acesso em: 19 out. 2018

BIOMAS DO MUNDO. TODAMATERIA. Disponível em:
<<https://www.todamateria.com.br/biomas-do-mundo/>>. Acesso em: 27 out. 2018

BIOMAS TERRESTRES. ESTUDOPRATICO. Disponível em:
<<https://www.estudopratico.com.br/biomas-terrestres/>>. Acesso em: 28 out. 2018

BIOMAS-BIODIVERSIDADE-ECOSSISTEMAS. Disponível em:
<http://www.outorga.com.br/pdf/Principais%20biomas%20do%20mundo.pdf>>. Acesso em: 28 out. 2018

DESENVOLVIMENTO DE JOGOS EDUCACIONAIS UTILIZANDO A BIBLIOTECA ALLEGRO. Disponível em:
<https://www.academia.edu/1433991/Desenvolvimento_de_Jogos_Educacionais_ut.pdf>. Acesso em: 18 out. 2018

EFEITOS SONOROS E EFEITOS SONOROS ROYALTY FREE. Disponível em:
<<https://www.partnersinrhyme.com/pir/PIRsfx.shtml>>. Acesso em: 14 nov. 2018

ESCREVENDO TEXTO USANDO TRUE TYPE. Disponível em:
<<http://www.rafaeltoledo.net/tutorial-allegro-5-4-escrevendo-texto-usando-fontes-true-type-ttf/>>. Acesso em: 14 nov. 2018

FLORESTA TEMPERADA. INFOESCOLA. Disponível em:
<<https://www.infoescola.com/biomas/floresta-temperada/>>. Acesso em: 28 out. 2018

INSTALANDO ALLEGRO NO DEV CPP. Disponível em:
<<https://www.tuliofaria.net/instalando-a-allegro-no-dev-c-dev-cpp/>>. Acesso em: 28 out. 2018

INTRODUÇÃO A PROGRAMAÇÃO EM C COM JOGOS 2D. Disponível em:
<<https://sites.google.com/a/liesenberg.biz/cjogos/home/recursos/allegro>>. Acesso em: 02 nov. 2018

JOGOS EM C UTILIZANDO A BIBLIOTECA ALLEGRO. Disponível em:
<<https://updatedcode.wordpress.com/2012/01/24/jogos-em-c-usando-a-biblioteca-allegro/>>. Acesso em: 28 out. 2018

MENU PARA GAMES EM C ALLEGRO. Disponível em:
<https://www.youtube.com/results?search_query=como+fazer+um+menu+allegro>. Acesso em: 03 nov. 2018

PAMPAS.UOL. Disponível em: <<https://www.estudopratico.com.br/pampas-fauna-flora-caracteristicas-e-problemas/>>. Acesso em: 28 out. 2018

PANTANAL. INFOESCOLA. Disponível em:
<<https://www.infoescola.com/biomas/pantanal/>>. Acesso em: 29 out. 2018

PRADARIAS. INFOESCOLA. Disponível em:
<<https://www.infoescola.com/geografia/pradarias-estepes/>>. Acesso em: 29 out. 2018

SONS URBANOS. Disponível em:
<http://audiocidades.utopia.org.br/biblioteca/biblioteca_sons_urbanos.html>. Acesso em: 15 nov. 2018

TRUNDA. INFOESCOLA. Disponível em: <<https://www.infoescola.com/biomas/tundra/>>. Acesso em: 29 out. 2018

TUTORIAL ALLEGRO, DA INSTALAÇÃO ATÉ A PROGRAMAÇÃO. Disponível em:
<<https://www.hardware.com.br/comunidade/tutorial-allegro/1035452/>>. Acesso em: 29 out. 2018

TUTORIAL ALLEGRO. Disponível em:
http://edcomjogos.dc.ufscar.br/tutoriais/tutorial_allegro.pdf. Acesso em: 01 nov. 2018

ANEXO A – CÓDIGO DO JOGO

```
#include <allegro.h>

volatile int exit_program;
void fecha_programa() { exit_program = TRUE; }    //Fechar o programa
END_OF_FUNCTION(fecha_programa)

volatile int ticks;
void tick_counter() { ticks++; }                //Trocar a janela
END_OF_FUNCTION(tick_counter)

volatile int timer;
void incrementa_timer(){timer++;}              //Cronômetro
END_OF_FUNCTION(incrementa_timer)

volatile int pontos;
void incrementa_pt(){pontos = pontos+100;}      // Pontuação
END_OF_FUNCTION(incrementa_pt)

void pt2(){pontos = pontos+50;}                // Pontuação
END_OF_FUNCTION(incrementa_pt)

enum{ SPLASHSCREEN, GAMEMENU, FASE1, FASE2, FASE3, FASE4, FASE5, FASE6,
FASE7, FASE8, FASE9, FASE10, ERROU, ACERTOU, TM_ESG, AJUDA, GAMESCREEN };
int screen_state;

SAMPLE *clique = load_wav("clique.wav");
SAMPLE* erro = load_wav("errou.wav");          //Variáveis globais
SAMPLE* certo = load_wav("certo.wav");

void init()
{
    allegro_init();
    install_timer();
    install_keyboard();
    install_mouse();
    install_sound(DIGI_AUTODETECT,MIDI_AUTODETECT,NULL);
    set_color_depth(32);
    set_gfx_mode(GFX_AUTODETECT_WINDOWED, 640, 480, 0, 0);
    set_window_title("Bio Logicas");

    exit_program = FALSE;
    LOCK_FUNCTION(fecha_programa);
    LOCK_VARIABLE(exit_program);
    set_close_button_callback(fecha_programa);

    ticks = 0;
    LOCK_FUNCTION(tick_counter);
    LOCK_VARIABLE(ticks);
    install_int_ex(tick_counter, BPS_TO_TIMER(60));

    pontos = 0;
    LOCK_FUNCTION(incrementa_pt);
    LOCK_VARIABLE(pontos);

    screen_state = GAMEMENU;
}
```

```

void fase1()
{
    int exit_screen = FALSE;
    int time = 0;

    ///BITMAPS
    BITMAP* buffer = create_bitmap(SCREEN_W, SCREEN_H);
    BITMAP *onca = load_bitmap("onca.bmp", NULL);
    BITMAP *arara = load_bitmap("arara.bmp", NULL);
    BITMAP *canguru = load_bitmap("canguru.bmp", NULL);
    BITMAP *pau_brasil = load_bitmap("chuva.bmp", NULL);
    BITMAP *veg_fechada = load_bitmap("veg_fechada.bmp", NULL);
    BITMAP *guarana = load_bitmap("guarana.bmp", NULL);
    BITMAP *cursor = load_bitmap("cursor.bmp", NULL);
    BITMAP *fundo = load_bitmap("fundo3.bmp", NULL);
    FONT *font14 = load_font("font14.pcx", NULL, NULL);
    FONT *fontass = load_font("fontass.pcx", NULL, NULL);
    FONT *font24 = load_font("font24.pcx", NULL, NULL);

    timer = 0;
    LOCK_FUNCTION(incrementa_timer);
    LOCK_VARIABLE(timer);
    install_int_ex(incrementa_timer, SECS_TO_TIMER(1)); //CRONÔMETRO//

    ///GAME LOOP
    while(!exit_program && !exit_screen)
    {
        while(ticks > 0 && !exit_program && !exit_screen)
        {
            if(key[KEY_ESC])
                exit_screen = TRUE;
            screen_state = GAMEMENU;

            if(timer>=10){
                exit_screen = TRUE;

                screen_state = TM_ESG;}

            if(timer>=7){
                if(time>30) time=0;}

            blit(fundo, buffer, 0, 0, 0, 0, 640, 480);
                if(time<20)

            blit(onca, buffer, 0, 0, 50, 130, 640, 480);
            blit(guarana, buffer, 0, 0, 240, 130, 640, 480);
            blit(canguru, buffer, 0, 0, 430, 130, 640, 480);
            blit(pau_brasil, buffer, 0, 0, 50, 260, 640, 480);
            blit(arara, buffer, 0, 0, 240, 260, 640, 480);
            blit(veg_fechada, buffer, 0, 0, 430, 260, 640, 480);

```

```

if(mouse_b & 1){
    if(mouse_x>430 && mouse_x<600 && mouse_y>130 && mouse_y<250){
        play_sample(certo, 255, 128, 1000, 0);
        if(timer<=5){
            incrementa_pt();}
        if(timer>5){
            pt2();}
        exit_screen = TRUE;
        screen_state = FASE2;}
    if(mouse_x>50 && mouse_x<415 && mouse_y>130 && mouse_y<375){
        exit_screen = TRUE;
        screen_state = ERROU;}
    if(mouse_x>428 && mouse_x<600 && mouse_y>255 && mouse_y<375){
        exit_screen = TRUE;
        screen_state = ERROU;}}

    draw_sprite(buffer, cursor, mouse_x-6, mouse_y);
    draw_sprite(screen, buffer, 0, 0);
    clear_to_color(buffer, makecol(255,255,255));

    time++;
    ticks--;
}
}
///GAME LOOP
while(!exit_program && !exit_screen)
{
    while(ticks > 0 && !exit_program && !exit_screen)
    {
        if(key[KEY_ENTER]){
            exit_screen = TRUE;
            play_sample(clique, 255, 128, 1000, 0);
            exit_screen = TRUE;
            screen_state = AJUDA;}

        if(key[KEY_ESC])
            fecha_programa();

        if(time > 30) time=0;

        pontos=0;
        blit(logo, buffer, 0, 0, 0, 0, 640,480);
        if(time < 20)

void gamemenu()
{
    int exit_screen = FALSE;
    int time = 0;

    ///BITMAPS
    BITMAP* buffer = create_bitmap(SCREEN_W, SCREEN_H);
    BITMAP* logo = load_bitmap("logo.bmp",NULL);
    FONT* font20 = load_font("font20.pcx",NULL,NULL);

```

```

    textprintf_centre_ex(buffer, font20, SCREEN_W/2, 330,
    makecol(255, 255, 255), -1, "Press enter to start");

    draw_sprite(screen, buffer, 0, 0);
    clear_to_color(buffer, makecol(255,255,255));

    time++;
    ticks--;
}
}

int main()
{
    init();
    SAMPLE *selva = load_wav("selva.wav");
    SAMPLE *aplausos = load_wav("aplausos.wav");
    SAMPLE *menu = load_wav("menu.wav");
    SAMPLE *fases = load_wav("fases.wav");

    while(!exit_program)
    {
        if(screen_state == FASE1){
            play_sample (fases, 150, 128, 1000, 1);
            fase1();}

        if(screen_state == GAMEMENU){
            stop_sample(fases);
            stop_sample(aplausos);
            stop_sample(erro);
            stop_sample(clique);
            play_sample (selva, 150, 128, 1000, 1);
            play_sample(menu, 255, 128, 1000, 1);
            gamemenu();}

        if(screen_state == FASE2){
            fase2();}

        if(screen_state == ERROU){
            stop_sample(fases);
            play_sample(erro, 255, 128, 1000, 0);
            erro();}

        if(screen_state == ACERTOU){
            stop_sample(selva);
            play_sample (aplausos, 255, 128, 1000, 0);
            acertou();}

        if(screen_state == TM_ESG){
            stop_sample(fases);
            tm_esg();}

        if(screen_state == AJUDA){
            stop_sample(menu);
            stop_sample(selva);
            ajuda();}
    }

    return 0;
}
END_OF_MAIN();

```