**Documentatie proiect PMP**

**Flappy Bird**

Arseniuc Anamaria

Grupa 30236, an 3, CTI

***Schema***

Elemente folosite:

* Arduino Uno
* LCD 2004 cu I2C integrat
* Breadboard
* 1 buton
* Fire mama-tata, tata-tata
* Cablu A-B pentru conectarea placutei la laptop

Conectare lcd la placa Conectare buton

GND->GND 2 fire, unul la GND si unul la pinul 2 de pe placa

VCC-> VCC

SDA->A4

SCL->A5

A picture containing text, indoor

Description automatically generated

A picture containing text

Description automatically generatedA picture containing calendar

Description automatically generatedA close - up of a circuit board

Description automatically generated with medium confidence

**Codul:**

Am facut totul in interiorul loop-ului si a setup-ului. Am doar o functie care se apeleaza la apasarea butonului (intrerupere exterioara).

Obstacolele le-am creat prin printarea pe lcd a caracterului ‘X’ de-a lungul axei Y, lasand doar un loc pentru a putea trece Flappy Bird-ul.

Pentru crearea Flappy-ului, cat si a monedei, am activat acele pozitii din pixel pentru a reda ‘imaginea’ vruta.

Ex.

A picture containing text, whiteboard

Description automatically generated

//includerea bibliotecii (pentru lcd-ul 20\*4 cu I2C integrat)

#include <LiquidCrystal\_I2C.h>

// seteaza adresa lcd-ului la '0x27' pentru lcd-ul cu 4 linii si 20 coloane

LiquidCrystal\_I2C lcd(0x27, 20, 4);

//bitii corespunzatori unui pixel 5\*8 pentru desenarea Flappy-ului

byte Flappy[] = {

B00000,

B01110,

B01101,

B00110,

B11110,

B01110,

B01100,

B00000

};

//bitii corespunzatori unui pixel pentru desenarea monedei (reprezentata de o 'stea')

byte star[] = {

B00000,

B00100,

B10101,

B01110,

B10101,

B00100,

B00000,

B00000

};

//pentru buton, folosim pinul 2 de pe placa

#define BTN\_PIN 2

//variabilele pentru pozitia locului liber pentru trecerea Flappy

int holePosY;

int holePosX;

//variabilele pentru retinerea pozitiei monedei

int moneyPosY;

int moneyPosX;

//variabilele pentru pozitia actuala a Flappy-ului

int flappyPosY;

int flappyPosX;

//retinerea scorului

int score = 0;

//variabila care se activeaza atunci cand jocul se termina

//(Flappy nu trece de obstacol)

bool isGameOver = false;

static bool buttonPushed = false;

//functia pentru intrerupere

void buttonPush() {

buttonPushed = true;

}

void setup() {

// initializarea si activarea luminii lcd-ului

lcd.init();

lcd.backlight();

// Activare buton

pinMode(BTN\_PIN, INPUT\_PULLUP);

digitalWrite(BTN\_PIN, HIGH);

//atasam intreruperii/butonului 2 functia buutonPush

attachInterrupt(0/\*PIN\_BUTTON\*/, buttonPush, RISING);

//crearea caracterului care reprezinta Flappy Bird-ul

lcd.createChar(0, Flappy);

lcd.createChar(1, star);

//setam pozitia Flappy Bird-ului la (2,1)

flappyPosY = 1;

flappyPosX = 2;

//setam pozitia initiala a monedei la (3,17)

moneyPosY = random(0,4);

moneyPosX = random(5,20);

//initializam pozitia locului liber

holePosY = random(0, 4);

holePosX = 16;

}

int lastTime = 0;

void loop() {

//daca jocul e in rulare

if (millis() - lastTime >= 900 && !isGameOver) {

//in cazul in care nu se apasa butonul, Flappy coboara cu o pozitie

if (buttonPushed == true) {

if (flappyPosY - 1 >= 0) {

flappyPosY--;

}

buttonPushed = false;

}

//daca e apasat butonul, Flappy Bird urca o pozitie

else {

if (flappyPosY + 1 <= 3) {

flappyPosY++;

}

}

lastTime = millis();

}

//odata la 1/4 dintr-o secunda (in cazul in care jocul nu s-a terminat)

//se fac actualizarile cadrului (afisarea obstacolelor, monezilor, Flappy-ului pe lcd)

if (millis() % 250 == 0 && !isGameOver) {

//jocul e terminat cand Flappy atinge obstacolul (NU ARE aceeasi pozitie ca si al gaurii)

if (flappyPosY != holePosY && flappyPosX == holePosX) {

isGameOver = true;

return;

}

//daca Flappy atinge moneda, i se adauga 20 pct la scorul final

if (flappyPosY == moneyPosY && flappyPosY == moneyPosX) {

score+= 20;

lcd.setCursor(17, 0);

lcd.print(score);

}

lcd.clear();

//atunci cand un obstacol ajunge in partea din stanga a lcd-ului,

// se introduce un alt obstacol, cat si o moneda noua

if (holePosX == 0) {

holePosY = random(0,4);

holePosX = random(15,20);

score++;

moneyPosY = random(4);

moneyPosX = random(11,20);

}

//afisarea obstacolelor (x -> pt obstacol,

// " " -> spatiu pentru locul prin care poate trece Flappy Bird

for (int i = 0; i < 4; i++) {

lcd.setCursor(holePosX, i);

if (i == holePosY) {

lcd.print(" ");

} else {

lcd.print("X");

}

}

//la fiecare 1/4 secunde, obstacolul si banii se deplaseaza la stanga

moneyPosX--;

holePosX--;

//afisarea caracterului specific monezii

lcd.setCursor(moneyPosX, moneyPosY);

lcd.write(byte(1));

//afisarea Flappy-ului

lcd.setCursor(flappyPosX, flappyPosY);

lcd.write(byte(0));

//afisarea scorului in partea dreapta-sus

lcd.setCursor(17, 0);

lcd.print(score);

}

//la terminarea jocului, cand Flappy a atins un obstacol,

//se afiseaza pe ecran punctajul obtinut

if (isGameOver) {

lcd.clear();

lcd.setCursor(14, 0);

lcd.print("Flappy");

lcd.setCursor(16, 1);

lcd.print("Bird");

lcd.setCursor(0, 0);

lcd.print("YOU LOST!");

lcd.setCursor(0, 2);

lcd.print("Your score: ");

lcd.setCursor(0, 3);

lcd.print(score);

lastTime = millis();

delay(10000);a

}

//terminare loop }

*Link filmulet youtube* : <https://youtu.be/bvWc7d-RIvk>

**Bibliografie:**

<http://www.mirceapaulmuresan.com/papers/PMP-RO3.pdf?fbclid=IwAR0K2I4_OMToFw4ITN6dna7h5xhzs7VV1G9Oj0JbVQLYjkWSqF1Uet4XGeQ>

<https://maxpromer.github.io/LCD-Character-Creator/>

https://www.instructables.com/Arduino-Button-with-no-resistor/