**MATHOPIA**

**Code:**

#include <LiquidCrystal.h>//LCD library

LiquidCrystal lcd(6,5,4,3,2,1);//lcd(RS,En,D4,D5,D6,D7)

#include<EEPROM.h>

//A0 pin is used to reset the board

#define PINtoRESET A0// connected to reset pin of Uno

// define inputs

#define ip\_1 7

#define ip\_2 8

#define ip\_3 9

#define ip\_4 10

// define outputs

#define op\_1 11

#define op\_2 12

#define op\_3 13

// Buzzer

#define BUZZER A1

// variable to store random number

int randNumber = 0;

unsigned long startMillis; //time, when a question is given

unsigned long currentMillis;//current time

const unsigned long period = 10000;// time to answer the question

// Array of questions

String problems[20] = {

"3 X 6 + 116 =", "19 - 34 X 61= ", "100 / 25 X 25= ", "99 - 56 X 225/5", "-3 + 2x = 11", "4x + 6 = -10" , "x + 9 = 18 - 2x", "2x + 6 = 4x - 2",

"-x-1 = 221 + 2x" , "15 + 5x = 0", "17x-12 = 114+3x", "2x-10 = 10-3x", "12x + 60 = 144", "-10x-19 = 19-8x", "6/2 + 7 X 4 =", "4 + 82X(30/5) =",

"3(125-?) = 3", "x% of 5=6% of 40", "616+ ? +333=255", "[6-{4-(8-6+3)}]"

};

//Array of real solutions

int solutions[20] = {

134, -2055, 100, -2421, 7, -4, 3, 4, -74, -3, 9, 4, 7, -19, 31, 496, 124, 48, -694, 7

};

//Array to store the typed solution

char keypad[10] = {1, 10, 10, 10, 10, 10, 10, 10, 10, 10};

//solution counter to be used in various functions

int solcounter = 0;

void setup() {

//uncomment this while uploading the code for the first time

/\*

EEPROM.write(0,0);

EEPROM.write(1,0);\*/

// if analog input pin A2 is unconnected, random analog

// noise will cause the call to randomSeed() to generate

// different seed numbers each time the sketch runs.

// randomSeed() will then shuffle the random function.

randomSeed(analogRead(A2));

pinMode(PINtoRESET, INPUT); // set A0 pin as input

digitalWrite(PINtoRESET, LOW); // if set high, board will reset

lcd.begin(16, 2);

//set pins as output

for (int i = op\_1; i <= op\_3; i++)

pinMode(i, OUTPUT);

//set pins as input

pinMode(ip\_1, INPUT\_PULLUP);

pinMode(ip\_2, INPUT\_PULLUP);

pinMode(ip\_3, INPUT\_PULLUP);

pinMode(ip\_4, INPUT\_PULLUP);

initiate\_Game();// call the function to initiate game

start\_Game();// call the function for the first question

}

void loop() {

for (int i = op\_1; i <= op\_3; i++)

digitalWrite(i, HIGH);

// start checking for the pressed button in each column one by one

column1();

column2();

column3();

// compare the time elapsed since the question is displayed

currentMillis = millis();

if ((currentMillis - startMillis) > period)

{ // if time elapsed is > the period specified

lcd.clear();

lcd.setCursor(0, 0);

lcd.print("Out of Time!");

lcd.setCursor(0, 1);

//print the user score

lcd.print("Your Score =");

lcd.print(solcounter);

if(solcounter>EEPROM.read(0)){

EEPROM.write(0,solcounter);

}

//this code prevents program from printing the "welcoming message" for 10 consecutive resets or shut down

if (EEPROM.read(1)>9){

EEPROM.write(1,0);

}else{

int x= EEPROM.read(1);

EEPROM.write(1,++x);

}

/\*

Play this tone sequence when the answer is wrong or player is out of time

\*/

tone(BUZZER, 440, 200);

delay(200);

// turn off tone function for pin A2:

noTone(BUZZER);

tone(BUZZER, 494, 500);

delay(500);

// turn off tone function for pin A2:

noTone(BUZZER);

// play a note on pin 8 for 300 ms:

tone(BUZZER, 523, 300);

delay(300);

delay(1000);

lcd.clear();

pinMode(PINtoRESET, OUTPUT);

} else;

}

// function to intiate the game

void initiate\_Game()

{

lcd.clear();

lcd.setCursor(0, 0);

lcd.print("Welcome to ");

lcd.setCursor(3, 1);

lcd.print(" Mathopia");

if (EEPROM.read(1)==0){

// scroll 13 positions (string length) to the left

// to move it offscreen left:

for (int positionCounter = 0; positionCounter < 13; positionCounter++) {

// scroll one position left:

lcd.scrollDisplayLeft();

// wait a bit:

delay(200);

}

// scroll 29 positions (string length + display length) to the right

// to move it offscreen right:

for (int positionCounter = 0; positionCounter < 29; positionCounter++) {

// scroll one position right:

lcd.scrollDisplayRight();

// wait a bit:

delay(200);

}

// scroll 16 positions (display length + string length) to the left

// to move it back to center:

for (int positionCounter = 0; positionCounter < 16; positionCounter++) {

// scroll one position left:

lcd.scrollDisplayLeft();

// wait a bit:

delay(200);

}

}

lcd.setCursor(0,0);

lcd.print("High. Score=");

lcd.print(EEPROM.read(0));

delay(2000);

for (int i = op\_1; i <= op\_3; i++)

digitalWrite(i, LOW);

int w = 0, x = 0, y = 0, z = 0;

do

{

w = digitalRead(ip\_1);

x = digitalRead(ip\_2);

y = digitalRead(ip\_3);

z = digitalRead(ip\_4);

lcd.setCursor(0, 0);

lcd.print(" Press to Start");

lcd.setCursor(0, 1);

lcd.print("+-\*");

lcd.setCursor(13, 1);

lcd.print("-+/");

} while ((w == 1 && x == 1 && y == 1 && z == 1));// continue printing if button is not pressed

// else go back to the loop

delay(500);

}

// function to display the questions

void start\_Game()

{

lcd.clear();

// print a random number from 0 to 20

randNumber = random(0, 20);

//lcd.print(problems[solcounter]);

lcd.print(problems[randNumber]);

lcd.setCursor(0, 1);

startMillis = millis();

}

// function to store the typed number in an array(keypad[])

void fill\_Array(int x)

{ // check for the empty element in the array

for (int i = 1; i < 10; i++)// element 0 is reserved for the sign

{

if (keypad[i] == 10)

{

keypad[i] = x;

i = 10;

}

else;

}

}

//this function finds the last element that is stored(typed) in the keypad[] array

void find\_Last\_Element()

{

int endElement = 0;

for (int i = 1; i < 10; i++)

{

if (keypad[i] == 10) {

endElement = --i;

i = 10;

}

else;

}

check\_Ans(endElement);

}

// this function checks the answer stored in array keypad[] to the real answer stored in solutions array

void check\_Ans(int endElement)

{

long int answer = 0; // initialise the answer to 0

// for loop to copy the typed answer from typed[] array to the variable answer

for (int i = endElement, x = 1; (i >= 1) && (x <= endElement); i--, x++)

{

if (x == 1) {

answer += keypad[i];

}

else {

int tempNum = 1;

for (int y = 1; y < x; y++ )

{

tempNum = tempNum \* 10;

}

answer += keypad[i] \* tempNum;

}

}// end of for loop

answer \*= keypad[0]; // check for the sign of the entered answer

// if else conditional statment is used to compare real answer to typed answer

//if (answer==solutions[solcounter])

if (answer == solutions[randNumber])

{ //if answer is correct

lcd.setCursor(0, 1);

lcd.print("Correct!");

/\*

Play this tone sequence when the answer is correct

\*/

tone(BUZZER, 523, 300);

delay(300);

// turn off tone function for pin A2:

noTone(BUZZER);

tone(BUZZER, 494, 500);

delay(500);

// turn off tone function for pin A2:

noTone(BUZZER);

tone(BUZZER, 440, 200);

delay(200);

delay(1000);

solcounter++;

if (solcounter == 21) {

lcd.clear();

lcd.setCursor(0, 0);

lcd.print("Congratulations!");

lcd.setCursor(1, 3);

lcd.print("Champion");

/\*

Play this tone sequence after the correct answer to the last question

\*/

int x = 0;

while (x < 4) {

tone(BUZZER, 523, 300);

delay(300);

// turn off tone function for pin A2:

noTone(BUZZER);

tone(BUZZER, 494, 500);

delay(500);

// turn off tone function for pin A2:

noTone(BUZZER);

tone(BUZZER, 440, 200);

delay(200);

// turn off tone function for pin A2:

noTone(BUZZER);

tone(BUZZER, 494, 500);

delay(500);

// turn off tone function for pin A2:

noTone(BUZZER);

x++;

}

delay(2000);

pinMode(PINtoRESET, OUTPUT);// reset the board

}

keypad[0] = 1;

for (int i = 1; i < 10; i++)

{

keypad[i] = 10;

}

start\_Game();// call this function to display next question

}

else { //if answer is incorrect

lcd.clear();

lcd.setCursor(0, 0);

lcd.print("Wrong Answer");

lcd.setCursor(0, 1);

//print the user score

lcd.print("Your Score =");

lcd.print(solcounter);

//if previous high score is less than current score

if(solcounter>EEPROM.read(0)){

EEPROM.write(0,solcounter);

}

//this code prevents program from printing the "welcoming message" for 10 consecutive resets or shut down

if (EEPROM.read(1)>9){

EEPROM.write(1,0);

}else{

int x= EEPROM.read(1);

EEPROM.write(1,++x);

}

/\*

Play this tone sequence when the answer is wrong or player is out of time

\*/

tone(BUZZER, 440, 200);

delay(200);

// turn off tone function for pin A2:

noTone(BUZZER);

tone(BUZZER, 494, 500);

delay(500);

// turn off tone function for pin A2:

noTone(BUZZER);

// play a note on pin 8 for 300 ms:

tone(BUZZER, 523, 300);

delay(300);

delay(1000);

lcd.clear();

pinMode(PINtoRESET, OUTPUT);// reset the board

}

}

//check for the button pressed in column 1

void column1()

{

digitalWrite(11, LOW);

int w = digitalRead(ip\_1);

if (w < 1) {

delay(10);

lcd.print("1");

fill\_Array(1);// fill the array

tone(BUZZER, 1906, 200);

delay(500);

} else ;

int x = digitalRead(ip\_2);

if (x < 1) {

delay(10);

lcd.print("4");

fill\_Array(4);

tone(BUZZER, 1979, 200);

delay(500);

} else ;

int y = digitalRead(ip\_3);

if (y < 1) {

delay(10);

lcd.print("7");

fill\_Array(7);

tone(BUZZER, 2061, 200);

delay(500);

} else ;

int z = digitalRead(ip\_4);

if (z < 1) {

delay(10);

lcd.print("-");

keypad[0] = -1;

tone(BUZZER, 2150, 200);

delay(500);

} else ;

for (int i = op\_1; i <= op\_3; i++)

digitalWrite(i, HIGH);

}

//check for the button pressed in column 2

void column2()

{

digitalWrite(12, LOW);

int w = digitalRead(ip\_1);

if (w < 1) {

delay(10);

lcd.print("2");

fill\_Array(2);

tone(BUZZER, 2033, 200);

delay(500);

} else ;

int x = digitalRead(ip\_2);

if (x < 1) {

delay(10);

lcd.print("5");

fill\_Array(5);

tone(BUZZER, 2106, 200);

delay(500);

} else ;

int y = digitalRead(ip\_3);

if (y < 1) {

delay(10);

lcd.print("8");

fill\_Array(8);

tone(BUZZER, 2188, 200);

delay(500);

} else ;

int z = digitalRead(ip\_4);

if (z < 1) {

delay(10);

lcd.print("0");

fill\_Array(0);

tone(BUZZER, 2277, 200);

delay(500);

} else ;

for (int i = op\_1; i <= op\_3; i++)

digitalWrite(i, HIGH);

}

//check for the button pressed in column 3

void column3()

{

digitalWrite(13, LOW);

int w = digitalRead(ip\_1);

if (w < 1) {

delay(10);

lcd.print("3");

fill\_Array(3);

tone(BUZZER, 2174, 200);

delay(500);

} else ;

int x = digitalRead(ip\_2);

if (x < 1) {

delay(10);

lcd.print("6");

fill\_Array(6);

tone(BUZZER, 2247, 200);

delay(500);

} else ;

int y = digitalRead(ip\_3);

if (y < 1) {

delay(10);

lcd.print("9");

fill\_Array(9);

tone(BUZZER, 2329, 200);

delay(500);

} else ;

int z = digitalRead(ip\_4);

if (z < 1) {

delay(10);

lcd.setCursor(0, 1);

lcd.print(" ");

find\_Last\_Element();

} else ;

}