#include <Keypad.h>

#include <LiquidCrystal.h>

#include <Adafruit\_NeoPixel.h>

#define PIAN 31

#define PIBN 32

#define PICN 33

#define PIDN 34

#define PIEN 35

#define PIFN 36

#define PIGN 37

#define PIHN 38

#define PIIN 39

#define PIJN 40

#define PIKN 41

#define PILN 42

#define PIMN 43

#define NUMPIXELS 13// number of neopixels in strip

Adafruit\_NeoPixel pixels31 = Adafruit\_NeoPixel(NUMPIXELS, PIAN, NEO\_GRB + NEO\_KHZ800);

Adafruit\_NeoPixel pixels32 = Adafruit\_NeoPixel(NUMPIXELS, PIBN, NEO\_GRB + NEO\_KHZ800);

Adafruit\_NeoPixel pixels33 = Adafruit\_NeoPixel(NUMPIXELS, PICN, NEO\_GRB + NEO\_KHZ800);

Adafruit\_NeoPixel pixels34 = Adafruit\_NeoPixel(NUMPIXELS, PIDN, NEO\_GRB + NEO\_KHZ800);

Adafruit\_NeoPixel pixels35 = Adafruit\_NeoPixel(NUMPIXELS, PIEN, NEO\_GRB + NEO\_KHZ800);

Adafruit\_NeoPixel pixels36 = Adafruit\_NeoPixel(NUMPIXELS, PIFN, NEO\_GRB + NEO\_KHZ800);

Adafruit\_NeoPixel pixels37 = Adafruit\_NeoPixel(NUMPIXELS, PIGN, NEO\_GRB + NEO\_KHZ800);

Adafruit\_NeoPixel pixels38 = Adafruit\_NeoPixel(NUMPIXELS, PIHN, NEO\_GRB + NEO\_KHZ800);

Adafruit\_NeoPixel pixels39 = Adafruit\_NeoPixel(NUMPIXELS, PIIN, NEO\_GRB + NEO\_KHZ800);

Adafruit\_NeoPixel pixels40 = Adafruit\_NeoPixel(NUMPIXELS, PIJN, NEO\_GRB + NEO\_KHZ800);

Adafruit\_NeoPixel pixels41 = Adafruit\_NeoPixel(NUMPIXELS, PIKN, NEO\_GRB + NEO\_KHZ800);

Adafruit\_NeoPixel pixels42 = Adafruit\_NeoPixel(NUMPIXELS, PILN, NEO\_GRB + NEO\_KHZ800);

Adafruit\_NeoPixel pixels43 = Adafruit\_NeoPixel(NUMPIXELS, PILN, NEO\_GRB + NEO\_KHZ800);

LiquidCrystal lcd(12, 11, 10, 9, A4, A5);

int delayval = 100; // timing delay in milliseconds

int key=1;

int redColor = 0;

int greenColor = 0;

int blueColor = 0;

const byte ROWS = 4;

const byte COLS = 4;

char keys[ROWS][COLS] = {

{ '1', '2', '3', 'A' },

{ '4', '5', '6', 'B' },

{ '7', '8', '9', 'C' },

{ '\*', '0', '#', 'D' }

};

byte colPins[COLS] = { 18,19,20,21 }; // Pins connected to C1, C2, C3, C4

byte rowPins[ROWS] = { A0,A1,16,17 }; // Pins connected to R1, R2, R3, R4

int LCDRow = 0;

Keypad keypad = Keypad(makeKeymap(keys), rowPins, colPins, ROWS, COLS);

void setup() {

**Serial**.begin(9600);

lcd.begin(16, 2);

lcd.setCursor(LCDRow, 0);

pixels31.begin();

pixels32.begin();

pixels33.begin();

pixels34.begin();

pixels35.begin();

pixels36.begin();

pixels37.begin();

pixels38.begin();

pixels39.begin();

pixels40.begin();

pixels41.begin();

pixels42.begin();

pixels43.begin();

}

void loop() {

char key = keypad.getKey();

if (key != NO\_KEY) {

lcd.print(key);

lcd.setCursor (++LCDRow, 0);

}

switch(key){

case 1: mypixel31();

mypixel39();

mypixel36();

mypixel34();

mypixel32();

break;

case 2: mypixel32();

mypixel33();

break;

case 3: mypixel32();

mypixel35();

break;

case 4: mypixel32();

mypixel34();

mypixel37();

break;

case 5: mypixel32();

mypixel34();

mypixel38();

break;

case 6: mypixel32();

mypixel34();

mypixel36();

mypixel40();

break;

case 7: mypixel32();

mypixel34();

mypixel36();

mypixel41();

break;

case 8: mypixel32();

mypixel34();

mypixel36();

mypixel39();

mypixel42();

break;

default:

break;

}

}

void mypixel31(){

pixels31.setPixelColor(0, pixels31.Color(255, 0, 0));//Sixers Colors

pixels31.setPixelColor(1, pixels31.Color(0, 0, 255));

pixels31.setPixelColor(2, pixels31.Color(255, 17, 95));//brithstone ruby

pixels31.setPixelColor(3, pixels31.Color(0, 150, 200));//eye light blue

pixels31.setPixelColor(4, pixels31.Color(165, 100, 75));//hair brownis

pixels31.show();

}

void mypixel32(){

pixels32.setPixelColor(0, pixels32.Color(255, 0, 0));//Sixers Colors

pixels32.setPixelColor(1, pixels32.Color(0, 0, 255));

pixels32.setPixelColor(2, pixels32.Color(255, 17, 95));//brithstone ruby

pixels32.setPixelColor(3, pixels32.Color(0, 150, 200));//eye light blue

pixels32.setPixelColor(4, pixels32.Color(165, 100, 75));//hair brownish

pixels32.show();

}

void mypixel33(){

pixels33.setPixelColor(0, pixels33.Color(355, 0, 0));//Sixers Colors

pixels33.setPixelColor(1, pixels33.Color(0, 0, 355));

pixels33.setPixelColor(2, pixels33.Color(355, 17, 95));//brithstone ruby

pixels33.setPixelColor(3, pixels33.Color(0, 150, 300));//eye light blue

pixels33.setPixelColor(4, pixels33.Color(165, 100, 75));//hair brownish

pixels33.show();

}

void mypixel34(){

pixels34.setPixelColor(0, pixels34.Color(255, 0, 0));//Sixers Colors

pixels34.setPixelColor(1, pixels34.Color(0, 0, 255));

pixels34.setPixelColor(2, pixels34.Color(255, 17, 95));//brithstone ruby

pixels34.setPixelColor(3, pixels34.Color(0, 150, 200));//eye light blue

pixels34.setPixelColor(4, pixels34.Color(165, 100, 75));//hair brownish

pixels34.show();

}

void mypixel35(){

pixels35.setPixelColor(0, pixels35.Color(555, 0, 0));//Sixers Colors

pixels35.setPixelColor(1, pixels35.Color(0, 0, 555));

pixels35.setPixelColor(2, pixels35.Color(555, 17, 95));//brithstone ruby

pixels35.setPixelColor(5, pixels35.Color(0, 150, 500));//eye light blue

pixels35.setPixelColor(4, pixels35.Color(165, 100, 75));//hair brownish

pixels35.show();

}

void mypixel36(){

pixels36.setPixelColor(0, pixels36.Color(255, 0, 0));//Sixers Colors

pixels36.setPixelColor(1, pixels36.Color(0, 0, 255));

pixels36.setPixelColor(2, pixels36.Color(255, 17, 95));//brithstone ruby

pixels36.setPixelColor(3, pixels36.Color(0, 150, 200));//eye light blue

pixels36.setPixelColor(4, pixels36.Color(165, 100, 75));//hair brownish

pixels36.show();

}

void mypixel37(){

pixels37.setPixelColor(0, pixels37.Color(255, 0, 0));//Sixers Colors

pixels37.setPixelColor(1, pixels37.Color(0, 0, 255));

pixels37.setPixelColor(2, pixels37.Color(255, 17, 95));//brithstone ruby

pixels37.setPixelColor(3, pixels37.Color(0, 150, 200));//eye light blue

pixels37.setPixelColor(4, pixels37.Color(175, 100, 75));//hair brownish

pixels37.show();

}

void mypixel38(){

pixels38.setPixelColor(0, pixels38.Color(255, 0, 0));//Sixers Colors

pixels38.setPixelColor(1, pixels38.Color(0, 0, 255));

pixels38.setPixelColor(2, pixels38.Color(255, 18, 95));//brithstone ruby

pixels38.setPixelColor(3, pixels38.Color(0, 150, 200));//eye light blue

pixels38.setPixelColor(4, pixels38.Color(185, 100, 85));//hair brownish

pixels38.show();

}

void mypixel39(){

pixels39.setPixelColor(0, pixels39.Color(255, 0, 0));//Sixers Colors

pixels39.setPixelColor(1, pixels39.Color(0, 0, 255));

pixels39.setPixelColor(2, pixels39.Color(255, 17, 95));//brithstone ruby

pixels39.setPixelColor(3, pixels39.Color(0, 150, 200));//eye light blue

pixels39.setPixelColor(4, pixels39.Color(195, 100, 75));//hair brownish

pixels39.show();

}

void mypixel40(){

pixels40.setPixelColor(0, pixels40.Color(255, 0, 0));//Sixers Colors

pixels40.setPixelColor(1, pixels40.Color(0, 0, 255));

pixels40.setPixelColor(2, pixels40.Color(255, 110, 95));//brithstone ruby

pixels40.setPixelColor(3, pixels40.Color(0, 150, 200));//eye light blue

pixels40.setPixelColor(4, pixels40.Color(1105, 100, 105));//hair brownish

pixels40.show();

}

void mypixel41(){

pixels41.setPixelColor(0, pixels41.Color(255, 0, 0));//Sixers Colors

pixels41.setPixelColor(1, pixels41.Color(0, 0, 255));

pixels41.setPixelColor(2, pixels41.Color(255, 111, 95));//brithstone ruby

pixels41.setPixelColor(3, pixels41.Color(0, 150, 200));//eye light blue

pixels41.setPixelColor(4, pixels41.Color(1115, 110, 115));//hair brownish

pixels41.show();

}

void mypixel42(){

pixels42.setPixelColor(0, pixels42.Color(255, 0, 0));//Sixers Colors

pixels42.setPixelColor(1, pixels42.Color(0, 0, 255));

pixels42.setPixelColor(2, pixels42.Color(255, 121, 95));//brithstone ruby

pixels42.setPixelColor(3, pixels42.Color(0, 150, 200));//eye light blue

pixels42.setPixelColor(4, pixels42.Color(1215, 122, 125));//hair brownish

pixels42.show();

}

void mypixel43(){

pixels43.setPixelColor(0, pixels43.Color(255, 0, 0));//Sixers Colors

pixels43.setPixelColor(1, pixels43.Color(0, 0, 255));

pixels43.setPixelColor(2, pixels43.Color(255, 131, 95));//brithstone ruby

pixels43.setPixelColor(3, pixels43.Color(0, 150, 200));//eye light blue

pixels43.setPixelColor(4, pixels43.Color(1315, 132, 135));//hair brownish

pixels43.show();

}