#include "mbed.h"

# define dp23 P0\_0

//# include "stdio.h"

# include "SPI\_TFT\_ILI9341.h"

# include "string"

# include "vector"

#include "stdlib.h"

//#include "Arial12x12.h"

//# include "font\_big.h"

#include "Arial12x12.h"

int highscore=0;

const int GORNJA\_GRANICA=25;

const int DONJA\_GRANICA=235;

const int DESNA\_GRANICA=315;

const int LIJEVA\_GRANICA=5;

AnalogIn VRx(PTC1);

AnalogIn VRy(PTC2);

AnalogIn VRx2(PTB2);

AnalogIn VRy2(PTB1);

//Taster za dzojstik

DigitalIn SW(PTB3);

//Taster za dzojstik 2

DigitalIn SW2(PTB0);

SPI\_TFT\_ILI9341 TFT(PTD2,PTD3,PTD1,PTD0,PTA20,PTD5,"TFT");

enum Kretanje{Gore=0,Dolje,Desno,Lijevo,NemaKretnje};

class Pozicija{

private:

int x;

int y;

public:

Pozicija(){

x=0;

y=0;

}

Pozicija(int xx,int yy){

x=xx;

y=yy;

}

int getX(){return x;}

int getY(){return y;}

void setX(int xx){x=xx;}

void setY(int yy){y=yy;}

};

Pozicija hrana;

Kretanje kretanjeDzojstika1(){

//TODO: 0.2 malo vise za stanej mirovanja

if(VRx < 1.0/3.0)return Lijevo;

else if(VRx> 2.0/3.0)return Desno;

else if(VRy <1.0/3.0)return Gore;

else if(VRy > 2.0/3.0)return Dolje;

return NemaKretnje;

}

Kretanje kretanjeDzojstika2(){

if(VRx2 < 1.0/3.0)return Lijevo;

else if(VRx2> 2.0/3.0)return Desno;

else if(VRy2 <1.0/3.0)return Gore;

else if(VRy2 > 2.0/3.0)return Dolje;

return NemaKretnje;

}

int velicinaZmije(vector<Pozicija>&snake){

for(int i=0;i<snake.size();i++){

if(snake[i].getX()==0)return i-1;

}

return 30;

}

void startnaPozicijaSnake1(vector<Pozicija>& snake){

//TODO: provjeriti pozicije

snake[0].setX(50);

snake[0].setY(110);

snake[1].setX(40);

snake[1].setY(110);

snake[2].setX(30);

snake[2].setY(110);

snake[3].setX(20);

snake[3].setY(110);

//TODO: trebalo bi da je tacno

}

void startnaPozicijaSnake2(vector<Pozicija>& snake){

snake[0].setX(270);

snake[0].setY(110);

snake[1].setX(280);

snake[1].setY(110);

snake[2].setX(290);

snake[2].setY(110);

snake[3].setX(300);

snake[3].setY(110);

}

Kretanje trenutnoKretanje(vector<Pozicija>&snake){

if(snake[0].getX()<snake[1].getX() &&

snake[0].getY()==snake[1].getY())return Lijevo;

if(snake[0].getX()>snake[1].getX() &&

snake[0].getY()==snake[1].getY())return Desno;

if(snake[0].getX()==snake[1].getX() &&

snake[0].getY()>snake[1].getY())return Dolje;

if(snake[0].getX()==snake[1].getX() &&

snake[0].getY()<snake[1].getY())return Gore;

return Gore;//TODO: nepotrebno

}

void pomjeriGlavu(vector<Pozicija>& snake,Kretanje trenutno){

int x=snake[0].getX();

int y=snake[0].getY();

if(trenutno==Gore){

snake[0].setY(y-10);

}

else if(trenutno==Dolje){

snake[0].setY(y+10); //TODO: gore dolje +-

}

else if(trenutno==Desno){

snake[0].setX(x+10);

}

else{

snake[0].setX(x-10);

}

//TODO: mozda ovdje printf staviti da provjerimo gdej se krece

}

void odnosDzojstikaITrenutnogKretanja(Kretanje &trenutno,vector<Pozicija>& snake){

Kretanje tren=trenutnoKretanje(snake);

if( (trenutno==Desno && tren==Lijevo) ||

(trenutno==Lijevo && tren==Desno) ||

(trenutno==Gore && tren==Dolje) ||

(trenutno==Dolje && tren==Gore) )

trenutno=tren;

}

void kretanjeZmije(vector<Pozicija>& snake,bool prva){

Kretanje trenutno;

if(prva==true)

trenutno=kretanjeDzojstika1();

else

trenutno=kretanjeDzojstika2();

//ako dzojstik miruje

if(trenutno==NemaKretnje)

trenutno=trenutnoKretanje(snake);

else

odnosDzojstikaITrenutnogKretanja(trenutno,snake); //zabrani ako npr. krene lijevo a zmija vec ide desno

Pozicija pomocna(snake[0].getX(),snake[0].getY());

pomjeriGlavu(snake,trenutno);

//velicina +1 zbog zadnje pozicije da si i ona krece da bi mogla brisati

int velicina=velicinaZmije(snake)+1;

for(int i=1;i<velicina;i++){

int x=snake[i].getX();

int y=snake[i].getY();

snake[i].setX(pomocna.getX());

snake[i].setY(pomocna.getY());

pomocna.setX(x);

pomocna.setY(y);

}

}

bool udarilaUOkvir(vector<Pozicija>& snake){

if(snake[0].getX()>=DESNA\_GRANICA || snake[0].getX()<=LIJEVA\_GRANICA

|| snake[0].getY()>=DONJA\_GRANICA ||snake[0].getY()<=GORNJA\_GRANICA)return true;

return false;

}

bool udarilaSamaSebe(vector<Pozicija>&snake){

int velicina=velicinaZmije(snake);

for(int i=2;i<velicina;i++){

if(snake[0].getX()==snake[i].getX()

&& snake[0].getY()==snake[i].getY())return true;

}

return false;

}

bool sudarileSeGlavama(vector<Pozicija> &snake,vector<Pozicija>&snake2){

if(snake[0].getX()==snake2[0].getX() &&

snake[0].getY()==snake2[0].getY())return true;

return false;

}

bool udarilaZmijaDrugu(vector<Pozicija> &snake,vector<Pozicija>&snake2){

int velicina=velicinaZmije(snake2);

for(int i=1;i<velicina;i++){

if(snake[0].getX()==snake2[i].getX() &&

snake[0].getY()==snake2[i].getY())return true;

}

return false;

}

void crtajZmiju1(vector<Pozicija>& snake){

int velicina=velicinaZmije(snake);

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

TFT.fillrect(snake[i].getX()-4,snake[i].getY()-4,snake[i].getX()+4,snake[i].getY()+4,Black);

}

// KOD ZA BRISANJE ZADNJE POZOCIJE REPA, TJ. DA NE OSTAVLJA REP ZA SOBOM

TFT.fillrect(snake[velicina].getX()-4,snake[velicina].getY()-4,snake[velicina].getX()+4,snake[velicina].getY()+4,Green);

}

void crtajZmiju2(vector<Pozicija>& snake){

int velicina=velicinaZmije(snake);

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

TFT.fillrect(snake[i].getX()-4,snake[i].getY()-4,snake[i].getX()+4,snake[i].getY()+4,Red);

}

TFT.fillrect(snake[velicina].getX()-4,snake[velicina].getY()-4,snake[velicina].getX()+4,snake[velicina].getY()+4,Green);

}

enum Ishod{PrvaIzgubila,Nerijeseno,DrugaIzgubila,Nista};

bool izgubila(vector<Pozicija> &snake1,vector<Pozicija> & snake2){

if(udarilaUOkvir(snake1) || udarilaSamaSebe(snake1) ||

udarilaZmijaDrugu(snake1,snake2))return true;

return false;

}

Ishod validirajKretnju(vector<Pozicija>& snake1,vector<Pozicija>& snake2){

if(izgubila(snake1,snake2) && izgubila(snake2,snake1))return Nerijeseno;

else if(sudarileSeGlavama(snake1,snake2))return Nerijeseno;

else if(udarilaZmijaDrugu(snake1,snake2))return PrvaIzgubila;

else if(udarilaZmijaDrugu(snake2,snake1))return DrugaIzgubila;

else if(izgubila(snake1,snake2))return PrvaIzgubila;

else if(izgubila(snake2,snake1))return DrugaIzgubila;

return Nista;

}

void novaHrana(vector<Pozicija>& snake1, vector<Pozicija>& snake2){

bool nema=true;

int x=0;

int y=0;

int vel1=velicinaZmije(snake1);

int vel2=velicinaZmije(snake2);

while(nema){

nema=false;

x=rand()%100;

x=x\*10;

x=x%300;

y=rand()%100;

y\*=10;

y=y%200;

if(y<30)

y+=30; //TODO: mozda 30 ali ne vjerujem

if(y>230)

y-=30;

if(x<10)

x+=10;

if(x>310)

x-=10;

for(int i=0;i<vel1;i++)

if(snake1[i].getX()==x && snake1[i].getY()==y)nema=true;

for(int i=0;i<vel2;i++)

if(snake2[i].getX()==x && snake2[i].getY()==y)nema=true;

}

hrana.setX(x);

hrana.setY(y);

}

void crtajHranu(Pozicija &h){

TFT.fillrect(h.getX()-4,h.getY()-4,h.getX()+4,h.getY()+4,Blue);

}

void brisiHranu(Pozicija &h){

TFT.fillrect(h.getX()-4,h.getY()-4,h.getX()+4,h.getY()+4,Green);

}

bool pojedenaHrana(Pozicija &prva,Pozicija &druga, Pozicija & h)

{

if(prva.getX()==h.getX())return true;

if(druga.getY()==h.getY())return true;

return false;

/\*

if(snake1[0].getX()==h.getX() && snake1[0].getY()==h.getY())

return true;

else if(snake2[0].getX()==h.getX() && snake2[0].getY()==h.getY())

return true;

return false;\*/

}

int score(vector<Pozicija>& snake){

return velicinaZmije(snake)-3; //jer tri su pocetna

}

enum Meni {Start,High};

void promjeni(Meni & meni){

if(meni==Start)meni=High;

else meni=Start;

}

void nacrtajMeni(Meni & meni){

if(meni==Start){

TFT.locate(100,100);

printf("Start "); //bijelim slovima nekako to uradi

TFT.locate(100,150);

printf("High score");

}

else{

TFT.locate(100,100);

printf("High score"); //bijelim slovima nekako to uradi

TFT.locate(100,150);

printf("Start ");

}

}

void crtajMenu(Meni & meni){

Kretanje kretanje=kretanjeDzojstika1();

if(kretanje==Gore || kretanje==Dolje){

promjeni(meni);

nacrtajMeni(meni);

}

}

int main() {

// Aktivacija pull - up otpornika

SW.mode(PullUp);

SW2.mode(PullUp);

//podesavanje display-a

TFT.claim ( stdout );

TFT.set\_orientation (1);

TFT.background ( Blue );

TFT.foreground ( White );

TFT.cls ();

TFT.set\_font (( unsigned char \*) Arial12x12 );

Meni meni=Start;

nacrtajMeni(meni);

while(1){

crtajMenu(meni);

if(SW==0 && meni==High){

TFT.locate(100,150);

printf(" ");

TFT.locate(178,100);

TFT.\_putc('0'+highscore);

wait(2);

TFT.locate(177,100);

TFT.\_putc(' ');

TFT.locate(180,100);

TFT.\_putc(' ');

nacrtajMeni(meni);

}

else if(SW==0 && meni==Start)

{

// TFT.set\_orientation (1);

TFT.background ( Green );

TFT.foreground ( White );

TFT.cls();

TFT.locate(100,150);

printf(" ");

TFT.locate(100,100);

printf(" ");

//displej je dimenzija 320\*240

TFT.line(LIJEVA\_GRANICA,GORNJA\_GRANICA,DESNA\_GRANICA,GORNJA\_GRANICA,Black); //x osa gore TODO: mozda staviti 0?

TFT.line(LIJEVA\_GRANICA,GORNJA\_GRANICA,LIJEVA\_GRANICA,DONJA\_GRANICA,Black); //y osa | lijeva

TFT.line(LIJEVA\_GRANICA,DONJA\_GRANICA,DESNA\_GRANICA,DONJA\_GRANICA,Black); //x osa dolje

TFT.line(DESNA\_GRANICA,GORNJA\_GRANICA,DESNA\_GRANICA,DONJA\_GRANICA,Black); //y osa desna |

//TFT.set\_font (( unsigned char \*) Arial12x12 );

//int i=0;

//6 piksela jedno slovo zauzima

vector<Pozicija>snake1(31);

vector<Pozicija>snake2(31);

startnaPozicijaSnake1(snake1);

startnaPozicijaSnake2(snake2);

crtajZmiju1(snake1);

crtajZmiju2(snake2);

novaHrana(snake1,snake2);

crtajHranu(hrana);

//ubaciti timer da radi brze

int vel1=score(snake1);

int vel2=score(snake2);

TFT.locate(5,10);

TFT.\_putc('S');

TFT.locate(11,10);

TFT.\_putc('c');

TFT.locate(17,10);

TFT.\_putc('o');

TFT.locate(23,10);

TFT.\_putc('r');

TFT.locate(29,10);

TFT.\_putc('e');

TFT.locate(35,10);

TFT.\_putc(':');

TFT.locate(41,10);

TFT.\_putc('0');

TFT.locate(220,10);

TFT.\_putc('S');

TFT.locate(226,10);

TFT.\_putc('c');

TFT.locate(232,10);

TFT.\_putc('o');

TFT.locate(238,10);

TFT.\_putc('r');

TFT.locate(244,10);

TFT.\_putc('e');

TFT.locate(250,10);

TFT.\_putc(':');

TFT.locate(256,10);

TFT.\_putc('0');

//printf("Score: %i",vel2);

//printf(" ");

while(1) {

if(snake1[0].getX()==hrana.getX() && snake1[0].getY()==hrana.getY()){

novaHrana(snake1,snake2);

crtajHranu(hrana);

int velicina=velicinaZmije(snake1);

snake1[velicina+1].setX(1); //samo sam pomjerio da ide dalje a kretanje ce obaviti svoje i crtaje

snake1[velicina+1].setY(1); //prividno dodao na repu ali zapravo sam dodao na glavi

//TFT.locate(5,10);

vel1++;

//printf("Score: %i",vel1);

TFT.locate(41,10);

TFT.\_putc('0'+vel1);

//printf("");

}

else if(snake2[0].getX()==hrana.getX() && snake2[0].getY()==hrana.getY()){

novaHrana(snake1,snake2);

crtajHranu(hrana);

int velicina=velicinaZmije(snake2);

snake2[velicina+1].setX(1); //samo sam pomjerio da ide dalje a kretanje ce obaviti svoje i crtaje

snake2[velicina+1].setY(1); //prividno dodao na repu ali zapravo sam dodao na glavi

//TFT.locate(220,10);

vel2++;

//printf("Score: %i",vel2);

TFT.locate(256,10);

TFT.\_putc('0'+vel2%10);

}

kretanjeZmije(snake1,true);

crtajZmiju1(snake1);

kretanjeZmije(snake2,false); //FALSE ZA DRUGU ZMIJU ZA DZOJSTIk

crtajZmiju2(snake2);

Ishod ishod=validirajKretnju(snake1,snake2);

if(ishod==PrvaIzgubila){

TFT.locate(100,40);

TFT.\_putc('P');

TFT.locate(106,40);

TFT.\_putc('l');

TFT.locate(112,40);

TFT.\_putc('a');

TFT.locate(118,40);

TFT.\_putc('y');

TFT.locate(124,40);

TFT.\_putc('e');

TFT.locate(130,40);

TFT.\_putc('r');

TFT.locate(136,40);

TFT.\_putc('2');

TFT.locate(142,40);

TFT.\_putc(' ');

TFT.locate(148,40);

TFT.\_putc('w');

TFT.locate(154,40);

TFT.\_putc('o');

TFT.locate(160,40);

TFT.\_putc('n');

TFT.fillcircle(snake1[0].getX(),snake1[0].getY(),2,Green);

if(vel2>vel1)vel1=vel2;

if(vel1>highscore)highscore=vel1;

break;

}

else if(ishod==DrugaIzgubila){

TFT.locate(100,40);

TFT.\_putc('P');

TFT.locate(106,40);

TFT.\_putc('l');

TFT.locate(112,40);

TFT.\_putc('a');

TFT.locate(118,40);

TFT.\_putc('y');

TFT.locate(124,40);

TFT.\_putc('e');

TFT.locate(130,40);

TFT.\_putc('r');

TFT.locate(136,40);

TFT.\_putc('1');

TFT.locate(142,40);

TFT.\_putc(' ');

TFT.locate(148,40);

TFT.\_putc('w');

TFT.locate(154,40);

TFT.\_putc('o');

TFT.locate(160,40);

TFT.\_putc('n');

TFT.fillcircle(snake2[0].getX(),snake2[0].getY(),2,Green);

if(vel2>vel1)vel1=vel2;

if(vel1>highscore)highscore=vel1;

break;

}

else if(ishod==Nerijeseno){

TFT.locate(100,40);

TFT.\_putc('D');

TFT.locate(106,40);

TFT.\_putc('r');

TFT.locate(112,40);

TFT.\_putc('a');

TFT.locate(118,40);

TFT.\_putc('w');

TFT.fillcircle(snake1[0].getX(),snake1[0].getY(),2,Green);

TFT.fillcircle(snake2[0].getX(),snake2[0].getY(),2,Green);

if(vel1>highscore)highscore=vel1; //podesavanje highscore

break;

}

wait(0.2);

}

wait(3);

TFT.background(Blue);

TFT.cls();

Meni meni=Start;

nacrtajMeni(meni);

}

// wait(0.1);

// }

wait(0.2);

}

return 0;

}