**Universidad Autónoma “Gabriel René Moreno”**

**Facultad de ingeniería en ciencias de la computación y telecomunicaciones**



***Programación 1***

**Universitario:** Una alma caritativa

**Carrera:** Ingeniería en sistemas

**Docente:** Evans Balcázar Veizaga

**Grupo:** XXX

**Registro:** Si

**Santa Cruz – Bolivia**

## ESTRUCTURA DEL DOCUMENTO DEL PROYECTO

1. INTRODUCCION
2. ANTECEDENTES Y JUSTIFICACION
3. EL PROBLEMA
   * SITUACION PROBLEMÁTICA
   * FORMULACION DEL PROBLEMA
4. OBJETIVOS:
   * OBJETIVO GENERAL
   * OBJETIVOS ESPECÍFICOS
5. METODOLOGÍA:
6. MARCO TEÓRICO:
7. DESARROLLO
8. RESULTADOS:
9. CONCLUSIONES:
10. RECOMENDACIONES
11. BIBLIOGRAFÍA CITADA:
12. ANEXOS:

**1.Introducción:**

El ratón y el laberinto es un proyecto que tiene como objetivo aplicar lo aprendido en toda la clase de INF120, para hacer esto se ocupan diversos conceptos aprendidos en los temas que se avanzó, ya sea matrices, archivo tipo, POO, etc. Ademas de esto el proyecto ayudara al estudiante a buscar, aprender y profundizar más en los temas avanzados para poder hacer un proyecto de calidad que cumpla con los estándares de lo aprendido.

Para hacer el proyecto se utilizó principalmente matrices como base del programa, pero también se aplicó otras cosas aprendidas, se utiliza vectores para hacer más eficiente el uso de la memoria y manejo del programa, y se tuvo unos resultados esperados que fueron óptimos.



**2.- Antecedentes y justificación:**

Este proyecto tiene como justificación mejorar la capacidad del estudiante en las matrices, y los archivos con tipo, planteando una serie de objetivos con la finalidad que se logre aprender y poner en práctica lo que se investiga, aportando así un conocimiento de tipo practico, y teórico.

**3.- Planteamiento del problema de investigación:**

El aburrimiento se define formalmente como “la falta de cosas interesantes para ver, escuchar o hacer”. Precisamente es este el problema el cual nuestro proyecto planea resolver: Un videojuego el cual pueda entretener y divertir a varias personas al mismo tiempo sin hacerlas perder el interés ni sentirse con falta de estimulación

**4.- Objetivos:**

El proyecto tiene una serie de objetivos por cumplir:

**4.1 Objetivo general. -**

En el presente proyecto se pretende desarrollar un juego de pensamiento lógico, en dos dimensiones que simule ratón tratando de salir de un laberinto. Un jugador controla en el juego un ratón moviéndolo verticalmente y horizontalmente. El objetivo consiste en que el jugador tiene que guiar al ratón hacia la salida del laberinto habiendo 3 niveles por completar.

**4.2 Objetivo específico. –**

* Implementar adecuadamente diferentes estructuras para **guardar** la información del juego.
* Utilizar las ventajas que nos ofrece un manejo apropiado de la memoria cuando se trabaja con **muchas Niveles de Juego**.
* Utilizar **Matrices de Enteros** para optimizar la **graficación de diferentes Laberintos** y uso de memoria.

**5. Metodología**

Para realizar este proyecto se usarán distintas librerías y recursos que ofrece lazarus:

**1.-Formularios:** Se utilizará un formulario para que el usuario pueda ver y controlar los movimientos del ratón desde su teclado, o mouse

**2.-Matrices:** Se utilizará matrices para armar los distintos niveles de los laberintos, para mantener una eficacia en la creación de los mismos y el manejo.

**3.-Graficadora:** Para hacer una interfaz más amigable y mejor controlable para el usuario se usaran gráficos ya sean imágenes para los niveles, mejor entorno, etc.

**4.-Archivos Tipo:** El sistema de guardado que tendrá el juego será gracias a los archivos con tipo creando un SaveFile cada vez que el usuario guarde una partida.

**6. Marco teórico:**

Para la creación del proyecto se necesitaba tener un marco teórico clave de los cuales fueron más importantes:

**-Matrices:** Una matriz es una tabla bidimensional de números ordenados, se llama matriz A de dimensión M\*n a un conjunto de números reales dispuestos en m filas (reglones) y n columnas

**-Pensamiento lógico:** Se necesita de pensamiento lógico para poder resolver un laberinto, el pensamiento lógico es aquel que se desprende de las relaciones entre los objetos y procede de la propia elaboración del individuo. Surge a través de la coordinación de las relaciones que previamente ha creado entre los objetos.

**7. Desarrollo**

Para el desarrollo nos centraremos en el desarrollo de:

**El formulario:**  El formulario consta de: un selector de nivel, botones para abrir, guardar el juego, y un botón para comenzar a jugar.

**Matrices:** Los niveles o generación de niveles fueron hechos a partir de matrices, el código principal para crear un nivel a partir de una matriz es :

f,c:integer;

LN1:Array[1..4,1..4]of integer =((1,0,1,1),(1,0,0,1),(1,1,0,0),(1,1,1,1));

LN2:Array[1..6,1..6]of integer =((1,0,1,1,1,1),(1,0,0,0,0,1),(1,1,1,1,0,1),(1,0,1,1,0,1),(1,0,0,0,0,1),(1,0,1,1,1,1));

LN3:Array[1..8,1..8]of integer =((1,0,1,1,1,1,1,1),(1,0,0,0,0,0,0,1),(1,0,1,1,0,1,0,1),(1,1,1,1,0,1,1,1),(1,0,0,0,0,1,0,0),(1,0,1,1,1,1,0,1),(1,0,0,0,0,0,0,1),(1,1,1,1,1,1,1,1));

begin

case niv of

1:begin //4x4

fils:=4;

cols:=4;

for f:=1 to fils do

for c:=1 to cols do

Lab[f,c]:=LN1[f,c];

Ini.fil:=posF;

Ini.col:=PosC;

Fin.fil:=4;

Fin.col:=3;

end;

2:begin //6x6

fils:=6;

cols:=6;

for f:=1 to fils do

for c:=1 to cols do

lab[f,c]:=LN2[f,c];

Ini.fil:=posF;

Ini.col:=PosC;

Fin.fil:=2;

Fin.col:=6;

end;

3:begin //8x8

fils:=8;

cols:=8;

for f:=1 to fils do

for c:=1 to cols do

lab[f,c]:=LN3[f,c];

Ini.fil:=posF;

Ini.col:=PosC;

Fin.fil:=8;

Fin.col:=5;

end;

end;

raton:=Ini;

dimcam:=0;

nivel:=niv;

Dibujar();

**Graficadora:** Para generar los graficos del escenario del laberinto usamos imagines colocadas en cada cuadro de la matriz:

case nivel of

1: begin

sndPlaySound('sounds/Start.wav', SND\_NODEFAULT Or SND\_ASYNC or SND\_FILENAME);

ctx.Caption:='Nivel :'+IntToStr(nivel);

ctx.Width:=450;

ctx.Height:=450;

dx:=80;

dy:=80;

for f:=1 to 5 do begin

for c:=1 to 5 do begin

img:=TImage.Create(ctx);

img.Parent:=ctx;

img.Picture.LoadFromFile('images/FondoGris.png');

img.Left:=f\*dx;

img.Top:=c\*dy;

img.Width:=dx;

img.Height:=dy;

img.Stretch:=true;

end;

end;

for f:=1 to fils do

for c:=1 to cols do

begin

img:=TImage.Create(ctx);

img.Parent:=ctx;

if EsEsquinaDer(f,c) then begin

img.Picture.LoadFromFile('images/EsquinaDer.png');

sw:=true;

end else if sw=False then begin

img.Picture.LoadFromFile('images/TierraDefinitivo.png');

end;

if EsEsquinaIZ(C,F) then begin

img.Picture.LoadFromFile('images/EsquinaIZ.png');

sw:=true;

end else if sw=False then begin

img.Picture.LoadFromFile('images/TierraDefinitivo.png');

end;

if Vertical(C,F) then begin

img.Picture.LoadFromFile('images/VerticalImagen.png');

sw:=true;

end else if sw=False then begin

img.Picture.LoadFromFile('images/TierraDefinitivo.png');

end;

if(lab[f,c]=1)then begin

img.Picture.LoadFromFile('images/Pastito.png')

end;

img.Left:=f\*dx;

img.Top:=c\*dy;

img.Width:=dx;

img.Height:=dy;

img.Stretch:=true;

sw:=False;

end;

img:=TImage.Create(ctx);

img.Parent:=ctx;

img.Picture.LoadFromFile('images/Ratoncito.png');

img.Left:=raton.col\*dx;

img.Top:=raton.fil\*dy;

img.Width:=dx-20;

img.Height:=dy-20;

img.Stretch:=true;

filAntigua:=2;

ColAntigua:=1;

end;

2: begin

sndPlaySound('sounds/Start.wav', SND\_NODEFAULT Or SND\_ASYNC or SND\_FILENAME);

ctx.Caption:='Nivel: '+IntToStr(nivel);

ctx.Width:=600;

ctx.Height:=600;

dx:=80;

dy:=80;

for f:=1 to 7 do begin

for c:=1 to 7 do begin

img:=TImage.Create(ctx);

img.Parent:=ctx;

img.Picture.LoadFromFile('images/FondoGris.png');

img.Left:=f\*dx;

img.Top:=c\*dy;

img.Width:=dx;

img.Height:=dy;

img.Stretch:=true;

end;

end;

for f:=1 to fils do

for c:=1 to cols do

begin

img:=TImage.Create(ctx);

img.Parent:=ctx;

if EsEsquinaDer(f,c) then begin

img.Picture.LoadFromFile('images/EsquinaDer.png');

sw:=true;

end else if sw=False then begin

img.Picture.LoadFromFile('images/TierraDefinitivo.png');

end;

if EsEsquinaIZ(C,F) then begin

img.Picture.LoadFromFile('images/EsquinaIZ.png');

sw:=true;

end else if sw=False then begin

img.Picture.LoadFromFile('images/TierraDefinitivo.png');

end;

if Vertical(c,F) then begin

img.Picture.LoadFromFile('images/VerticalImagen.png');

sw:=true;

end else if sw=False then begin

img.Picture.LoadFromFile('images/TierraDefinitivo.png');

end;

if EsEsquinaDerDown(c,F) then begin

img.Picture.LoadFromFile('images/EsquinaDerDown.png');

sw:=true;

end else if sw=False then begin

img.Picture.LoadFromFile('images/TierraDefinitivo.png');

end;

if CentroSuperior(C,F) then begin

img.Picture.LoadFromFile('images/GrassUp.png');

sw:=true;

end else if sw=False then begin

img.Picture.LoadFromFile('images/TierraDefinitivo.png');

end;

if IzquierdaEncerrado(C,F) then begin

img.Picture.LoadFromFile('images/IzquierdaCerrado.png');

sw:=true;

end else if sw=False then begin

img.Picture.LoadFromFile('images/TierraDefinitivo.png');

end;

if(lab[f,c]=1)then begin

img.Picture.LoadFromFile('images/Pastito.png')

end;

img.Left:=f\*dx;

img.Top:=c\*dy;

img.Width:=dx;

img.Height:=dy;

img.Stretch:=true;

sw:=False;

end;

img:=TImage.Create(ctx);

img.Parent:=ctx;

img.Picture.LoadFromFile('images/Ratoncito.png');

img.Left:=raton.col\*dx;

img.Top:=raton.fil\*dy;

img.Width:=dx-20;

img.Height:=dy-20;

img.Stretch:=true;

filAntigua:=2;

ColAntigua:=1;

end;

3: begin

sndPlaySound('sounds/Start.wav', SND\_NODEFAULT Or SND\_ASYNC or SND\_FILENAME);

ctx.Caption:='Nivel: '+IntToStr(nivel);

ctx.Width:=800;

ctx.Height:=800;

dx:=80;

dy:=80;

for f:=1 to fils do begin

for c:=1 to cols do begin

img:=TImage.Create(ctx);

img.Parent:=ctx;

img.Picture.LoadFromFile('images/FondoGris.png');

img.Left:=f\*dx;

img.Top:=c\*dy;

img.Width:=dx;

img.Height:=dy;

img.Stretch:=true;

end;

end;

for f:=1 to fils do

for c:=1 to cols do

begin

img:=TImage.Create(ctx);

img.Parent:=ctx;

if EsEsquinaDer(C,f) then begin

img.Picture.LoadFromFile('images/EsquinaDer.png');

sw:=true;

end else if sw=False then begin

img.Picture.LoadFromFile('images/TierraDefinitivo.png');

end;

if EsEsquinaIZ(C,F) then begin

img.Picture.LoadFromFile('images/EsquinaIZ.png');

sw:=true;

end else if sw=False then begin

img.Picture.LoadFromFile('images/TierraDefinitivo.png');

end;

if Vertical(C,F) then begin

img.Picture.LoadFromFile('images/VerticalImagen.png');

sw:=true;

end else if sw=False then begin

img.Picture.LoadFromFile('images/TierraDefinitivo.png');

end;

if EsEsquinaDerDown(c,F) then begin

img.Picture.LoadFromFile('images/EsquinaDerDown.png');

sw:=true;

end else if sw=False then begin

img.Picture.LoadFromFile('images/TierraDefinitivo.png');

end;

if CentroSuperior(C,F) then begin

img.Picture.LoadFromFile('images/GrassUp.png');

sw:=true;

end else if sw=False then begin

img.Picture.LoadFromFile('images/TierraDefinitivo.png');

end;

if EsquinaIZUp(C,F) then begin

img.Picture.LoadFromFile('images/EsquinaIzSup.png');

sw:=true;

end else if sw=False then begin

img.Picture.LoadFromFile('images/TierraDefinitivo.png');

end;

if IzquierdaOnly(C,F) then begin

img.Picture.LoadFromFile('images/IzquierdaOnly.png');

sw:=true;

end else if sw=False then begin

img.Picture.LoadFromFile('images/TierraDefinitivo.png');

end;

if(lab[f,c]=1)then begin

img.Picture.LoadFromFile('images/Pastito.png')

end;

img.Left:=f\*dx;

img.Top:=c\*dy;

img.Width:=dx;

img.Height:=dy;

img.Stretch:=true;

sw:=False;

end;

img:=TImage.Create(ctx);

img.Parent:=ctx;

img.Picture.LoadFromFile('images/Ratoncito.png');

img.Left:=raton.col\*dx;

img.Top:=raton.fil\*dy;

img.Width:=dx-20;

img.Height:=dy-20;

img.Stretch:=true;

filAntigua:=2;

ColAntigua:=1;

end;

end;

ctx.Show;

**Archivos con tipo**: Se usará archivos con tipo para poder guardar el juego en cualquier parte que el usuario desee:

self.crear;

reg.fil:=raton.fil;

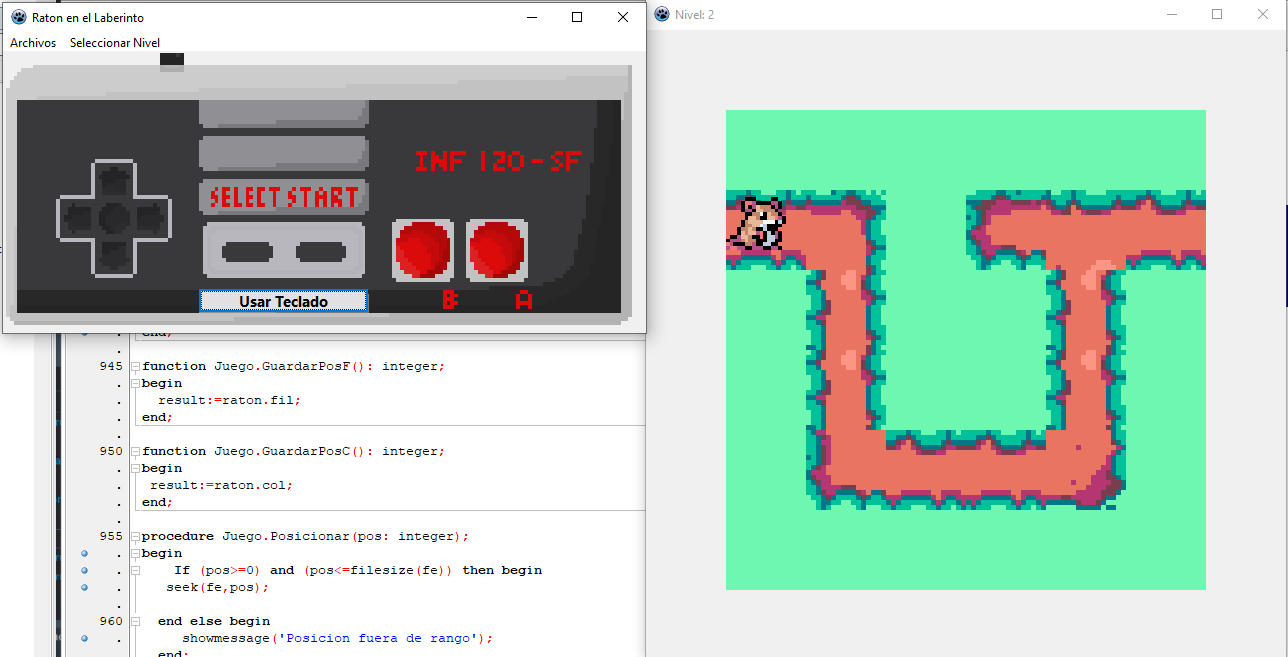
reg.col:=raton.col;

reg.nivelActual:=nivel;

self.escribir(reg);

self.cerrar;

**8.Resultados**

Los resultados del proyecto fueron muy bien, pese a los problemas que se plantearon mientras se desarrollaba pude arreglarlos todos cumpliendo así con cada objetivo específico que se enmarcaba, el proyecto está terminado es un buen juego para que las personas sanen el aburrimiento que a veces ocasiona estar en la cuarentena.

**9. Conclusiones y recomendaciones**

He concluido en varias cosas ya que aprendí muchas al hacer este proyecto, concluí que las matrices tienen más funciones además de ser solo para cálculos matemáticos, o que el sistema de guardado se puede aplicar eficientemente en varias cosas como un juego, las recomendaciones serian que a la hora de hacer estos proyectos seamos curiosos e investiguemos para poder hacer un buen proyecto.

**10. Bibliografía citada**

<https://www.lawebdelprogramador.com/foros/Delphi/index1.html>

- Introducción a la POO. Francisco Morero 1999-2000, Grupo EIDOS

-La cara oculta de Delphi 4, Ian Marteens 2005, Editorial Félix Varela

-https://stackoverflow.com

-https://en.delphipraxis.net

**Código fuente del proyecto (JuegoU)**

unit ujuego;

{$mode objfpc}{$H+}

interface

uses

Classes, SysUtils, Graphics, ExtCtrls, Dialogs, flab, MMSystem;

const

MAX\_F=25;

MAX\_C=25;

Arriba=1;

Abajo=2;

Izquierda=3;

Derecha=4;

type

Direccion=integer;

Posicion=record

fil,col,NivelActual:integer;

DireccionMirada:integer;

end;

{ Juego }

Juego=class

private

Fe:file of posicion;

Lab:array[1..MAX\_F,1..MAX\_C]of integer;

Fils,Cols:integer;

FilAntigua,ColAntigua:integer;

nom:string[120];

Est:integer;

Ini,Fin,raton:posicion;

Cam:array[1..MAX\_F\*MAX\_F]of posicion;

dimcam,nivel:integer;

//ctx es para interactuar con el Formulario

Ctx:TLab;

public

constructor Crear(Form:TLab);

procedure CrearLab(niv,PosF,PosC:integer);

procedure Automatico();

procedure Dibujar();

procedure DibujarRaton(DireccioMirando:integer);

procedure MoverRaton(dir:Direccion);

procedure setnom(n:string);

function getnom():string;

procedure crear();

procedure abrir();

procedure escribir(reg:posicion);

procedure leer(var reg:posicion);

procedure cerrar;

function FinDeArchivo():boolean;

procedure cargar();

Function GuardarPosF():integer;

Function GuardarPosC():integer;

Procedure Posicionar(pos:integer);

function EsEsquinaDer(F,C:integer):boolean;

Function EsEsquinaIz(F,c:integer):boolean;

Function Vertical(f,c:integer):boolean;

function EsEsquinaDerDown(f,c:integer):boolean;

function CentroSuperior(f,c:integer):boolean;

function EsquinaIzUp(f,c:integer):boolean;

Function IzquierdaOnly(f,c:integer):boolean;

function IzquierdaEncerrado(f,c:integer):boolean;

function GetNivel():integer;

end;

implementation

{ Juego }

constructor Juego.Crear(Form: TLab);

var

f,c:integer;

begin

fils:=0;

cols:=0;

for f:=1 to MAX\_F do

for c:=1 to MAX\_C do

Lab[f,c]:=-1;

ctx:=Form;

for f:=1 to MAX\_F\*MAX\_C do

begin

Cam[f].fil:=-1;

Cam[f].col:=-1;

end;

Ini.fil:=0;

Ini.col:=0;

Fin.fil:=0;

Fin.col:=0;

nom:='SaveFile.dat';

est:=0;

end;

procedure Juego.CrearLab(niv,PosF,PosC: integer);

var

f,c:integer;

LN1:Array[1..4,1..4]of integer =((1,0,1,1),(1,0,0,1),(1,1,0,0),(1,1,1,1));

LN2:Array[1..6,1..6]of integer =((1,0,1,1,1,1),(1,0,0,0,0,1),(1,1,1,1,0,1),(1,0,1,1,0,1),(1,0,0,0,0,1),(1,0,1,1,1,1));

LN3:Array[1..8,1..8]of integer =((1,0,1,1,1,1,1,1),(1,0,0,0,0,0,0,1),(1,0,1,1,0,1,0,1),(1,1,1,1,0,1,1,1),(1,0,0,0,0,1,0,0),(1,0,1,1,1,1,0,1),(1,0,0,0,0,0,0,1),(1,1,1,1,1,1,1,1));

begin

case niv of

1:begin //4x4

fils:=4;

cols:=4;

for f:=1 to fils do

for c:=1 to cols do

Lab[f,c]:=LN1[f,c];

Ini.fil:=posF;

Ini.col:=PosC;

Fin.fil:=4;

Fin.col:=3;

end;

2:begin //6x6

fils:=6;

cols:=6;

for f:=1 to fils do

for c:=1 to cols do

lab[f,c]:=LN2[f,c];

Ini.fil:=posF;

Ini.col:=PosC;

Fin.fil:=2;

Fin.col:=6;

end;

3:begin //8x8

fils:=8;

cols:=8;

for f:=1 to fils do

for c:=1 to cols do

lab[f,c]:=LN3[f,c];

Ini.fil:=posF;

Ini.col:=PosC;

Fin.fil:=8;

Fin.col:=5;

end;

end;

raton:=Ini;

dimcam:=0;

nivel:=niv;

Dibujar();

end;

procedure Juego.Automatico();

begin

//aqui debe ir codigo de juego automatico

case nivel of

1 : BEGIN

MoverRaton(derecha);

showmessage('F: '+inttostr(raton.fil)+', ' +'C: '+inttostr(raton.col));

MoverRaton(Abajo);

showmessage('F: '+inttostr(raton.fil)+', ' +'C: '+inttostr(raton.col));

MoverRaton(derecha);

showmessage('F: '+inttostr(raton.fil)+', ' +'C: '+inttostr(raton.col));

MoverRaton(Abajo);

end;

2: begin

MoverRaton(derecha);

showmessage('F: '+inttostr(raton.fil)+', ' +'C: '+inttostr(raton.col));

MoverRaton(Abajo);

showmessage('F: '+inttostr(raton.fil)+', ' +'C: '+inttostr(raton.col));

MoverRaton(abajo);

showmessage('F: '+inttostr(raton.fil)+', ' +'C: '+inttostr(raton.col));

MoverRaton(abajo);

showmessage('F: '+inttostr(raton.fil)+', ' +'C: '+inttostr(raton.col));

MoverRaton(derecha);

showmessage('F: '+inttostr(raton.fil)+', ' +'C: '+inttostr(raton.col));

MoverRaton(derecha);

showmessage('F: '+inttostr(raton.fil)+', ' +'C: '+inttostr(raton.col));

MoverRaton(derecha);

showmessage('F: '+inttostr(raton.fil)+', ' +'C: '+inttostr(raton.col));

MoverRaton(arriba);

showmessage('F: '+inttostr(raton.fil)+', ' +'C: '+inttostr(raton.col));

MoverRaton(arriba);

showmessage('F: '+inttostr(raton.fil)+', ' +'C: '+inttostr(raton.col));

MoverRaton(arriba);

showmessage('F: '+inttostr(raton.fil)+', ' +'C: '+inttostr(raton.col));

MoverRaton(derecha);

end;

3: begin

MoverRaton(derecha);

showmessage('F: '+inttostr(raton.fil)+', ' +'C: '+inttostr(raton.col));

MoverRaton(Abajo);

showmessage('F: '+inttostr(raton.fil)+', ' +'C: '+inttostr(raton.col));

MoverRaton(abajo);

showmessage('F: '+inttostr(raton.fil)+', ' +'C: '+inttostr(raton.col));

MoverRaton(abajo);

showmessage('F: '+inttostr(raton.fil)+', ' +'C: '+inttostr(raton.col));

MoverRaton(derecha);

showmessage('F: '+inttostr(raton.fil)+', ' +'C: '+inttostr(raton.col));

MoverRaton(derecha);

showmessage('F: '+inttostr(raton.fil)+', ' +'C: '+inttostr(raton.col));

MoverRaton(derecha);

showmessage('F: '+inttostr(raton.fil)+', ' +'C: '+inttostr(raton.col));

MoverRaton(arriba);

showmessage('F: '+inttostr(raton.fil)+', ' +'C: '+inttostr(raton.col));

MoverRaton(arriba);

showmessage('F: '+inttostr(raton.fil)+', ' +'C: '+inttostr(raton.col));

MoverRaton(arriba);

showmessage('F: '+inttostr(raton.fil)+', ' +'C: '+inttostr(raton.col));

MoverRaton(derecha);

showmessage('F: '+inttostr(raton.fil)+', ' +'C: '+inttostr(raton.col));

MoverRaton(derecha);

showmessage('F: '+inttostr(raton.fil)+', ' +'C: '+inttostr(raton.col));

MoverRaton(Abajo);

showmessage('F: '+inttostr(raton.fil)+', ' +'C: '+inttostr(raton.col));

MoverRaton(abajo);

showmessage('F: '+inttostr(raton.fil)+', ' +'C: '+inttostr(raton.col));

MoverRaton(abajo);

showmessage('F: '+inttostr(raton.fil)+', ' +'C: '+inttostr(raton.col));

MoverRaton(abajo);

showmessage('F: '+inttostr(raton.fil)+', ' +'C: '+inttostr(raton.col));

MoverRaton(abajo);

showmessage('F: '+inttostr(raton.fil)+', ' +'C: '+inttostr(raton.col));

MoverRaton (izquierda);

showmessage('F: '+inttostr(raton.fil)+', ' +'C: '+inttostr(raton.col));

MoverRaton (izquierda);

showmessage('F: '+inttostr(raton.fil)+', ' +'C: '+inttostr(raton.col));

MoverRaton(Abajo);

end;

end;

end;

procedure Juego.Dibujar();

var

f,c,dx,dy:integer;

img:TImage;

sw:boolean;

begin

case nivel of

1: begin

sndPlaySound('sounds/Start.wav', SND\_NODEFAULT Or SND\_ASYNC or SND\_FILENAME);

ctx.Caption:='Nivel :'+IntToStr(nivel);

ctx.Width:=450;

ctx.Height:=450;

dx:=80;

dy:=80;

for f:=1 to 5 do begin

for c:=1 to 5 do begin

img:=TImage.Create(ctx);

img.Parent:=ctx;

img.Picture.LoadFromFile('images/FondoGris.png');

img.Left:=f\*dx;

img.Top:=c\*dy;

img.Width:=dx;

img.Height:=dy;

img.Stretch:=true;

end;

end;

for f:=1 to fils do

for c:=1 to cols do

begin

img:=TImage.Create(ctx);

img.Parent:=ctx;

if EsEsquinaDer(f,c) then begin

img.Picture.LoadFromFile('images/EsquinaDer.png');

sw:=true;

end else if sw=False then begin

img.Picture.LoadFromFile('images/TierraDefinitivo.png');

end;

if EsEsquinaIZ(C,F) then begin

img.Picture.LoadFromFile('images/EsquinaIZ.png');

sw:=true;

end else if sw=False then begin

img.Picture.LoadFromFile('images/TierraDefinitivo.png');

end;

if Vertical(C,F) then begin

img.Picture.LoadFromFile('images/VerticalImagen.png');

sw:=true;

end else if sw=False then begin

img.Picture.LoadFromFile('images/TierraDefinitivo.png');

end;

if(lab[f,c]=1)then begin

img.Picture.LoadFromFile('images/Pastito.png')

end;

img.Left:=f\*dx;

img.Top:=c\*dy;

img.Width:=dx;

img.Height:=dy;

img.Stretch:=true;

sw:=False;

end;

img:=TImage.Create(ctx);

img.Parent:=ctx;

img.Picture.LoadFromFile('images/Ratoncito.png');

img.Left:=raton.col\*dx;

img.Top:=raton.fil\*dy;

img.Width:=dx-20;

img.Height:=dy-20;

img.Stretch:=true;

filAntigua:=2;

ColAntigua:=1;

end;

2: begin

sndPlaySound('sounds/Start.wav', SND\_NODEFAULT Or SND\_ASYNC or SND\_FILENAME);

ctx.Caption:='Nivel: '+IntToStr(nivel);

ctx.Width:=600;

ctx.Height:=600;

dx:=80;

dy:=80;

for f:=1 to 7 do begin

for c:=1 to 7 do begin

img:=TImage.Create(ctx);

img.Parent:=ctx;

img.Picture.LoadFromFile('images/FondoGris.png');

img.Left:=f\*dx;

img.Top:=c\*dy;

img.Width:=dx;

img.Height:=dy;

img.Stretch:=true;

end;

end;

for f:=1 to fils do

for c:=1 to cols do

begin

img:=TImage.Create(ctx);

img.Parent:=ctx;

if EsEsquinaDer(f,c) then begin

img.Picture.LoadFromFile('images/EsquinaDer.png');

sw:=true;

end else if sw=False then begin

img.Picture.LoadFromFile('images/TierraDefinitivo.png');

end;

if EsEsquinaIZ(C,F) then begin

img.Picture.LoadFromFile('images/EsquinaIZ.png');

sw:=true;

end else if sw=False then begin

img.Picture.LoadFromFile('images/TierraDefinitivo.png');

end;

if Vertical(c,F) then begin

img.Picture.LoadFromFile('images/VerticalImagen.png');

sw:=true;

end else if sw=False then begin

img.Picture.LoadFromFile('images/TierraDefinitivo.png');

end;

if EsEsquinaDerDown(c,F) then begin

img.Picture.LoadFromFile('images/EsquinaDerDown.png');

sw:=true;

end else if sw=False then begin

img.Picture.LoadFromFile('images/TierraDefinitivo.png');

end;

if CentroSuperior(C,F) then begin

img.Picture.LoadFromFile('images/GrassUp.png');

sw:=true;

end else if sw=False then begin

img.Picture.LoadFromFile('images/TierraDefinitivo.png');

end;

if IzquierdaEncerrado(C,F) then begin

img.Picture.LoadFromFile('images/IzquierdaCerrado.png');

sw:=true;

end else if sw=False then begin

img.Picture.LoadFromFile('images/TierraDefinitivo.png');

end;

if(lab[f,c]=1)then begin

img.Picture.LoadFromFile('images/Pastito.png')

end;

img.Left:=f\*dx;

img.Top:=c\*dy;

img.Width:=dx;

img.Height:=dy;

img.Stretch:=true;

sw:=False;

end;

img:=TImage.Create(ctx);

img.Parent:=ctx;

img.Picture.LoadFromFile('images/Ratoncito.png');

img.Left:=raton.col\*dx;

img.Top:=raton.fil\*dy;

img.Width:=dx-20;

img.Height:=dy-20;

img.Stretch:=true;

filAntigua:=2;

ColAntigua:=1;

end;

3: begin

sndPlaySound('sounds/Start.wav', SND\_NODEFAULT Or SND\_ASYNC or SND\_FILENAME);

ctx.Caption:='Nivel: '+IntToStr(nivel);

ctx.Width:=800;

ctx.Height:=800;

dx:=80;

dy:=80;

for f:=1 to fils do begin

for c:=1 to cols do begin

img:=TImage.Create(ctx);

img.Parent:=ctx;

img.Picture.LoadFromFile('images/FondoGris.png');

img.Left:=f\*dx;

img.Top:=c\*dy;

img.Width:=dx;

img.Height:=dy;

img.Stretch:=true;

end;

end;

for f:=1 to fils do

for c:=1 to cols do

begin

img:=TImage.Create(ctx);

img.Parent:=ctx;

if EsEsquinaDer(C,f) then begin

img.Picture.LoadFromFile('images/EsquinaDer.png');

sw:=true;

end else if sw=False then begin

img.Picture.LoadFromFile('images/TierraDefinitivo.png');

end;

if EsEsquinaIZ(C,F) then begin

img.Picture.LoadFromFile('images/EsquinaIZ.png');

sw:=true;

end else if sw=False then begin

img.Picture.LoadFromFile('images/TierraDefinitivo.png');

end;

if Vertical(C,F) then begin

img.Picture.LoadFromFile('images/VerticalImagen.png');

sw:=true;

end else if sw=False then begin

img.Picture.LoadFromFile('images/TierraDefinitivo.png');

end;

if EsEsquinaDerDown(c,F) then begin

img.Picture.LoadFromFile('images/EsquinaDerDown.png');

sw:=true;

end else if sw=False then begin

img.Picture.LoadFromFile('images/TierraDefinitivo.png');

end;

if CentroSuperior(C,F) then begin

img.Picture.LoadFromFile('images/GrassUp.png');

sw:=true;

end else if sw=False then begin

img.Picture.LoadFromFile('images/TierraDefinitivo.png');

end;

if EsquinaIZUp(C,F) then begin

img.Picture.LoadFromFile('images/EsquinaIzSup.png');

sw:=true;

end else if sw=False then begin

img.Picture.LoadFromFile('images/TierraDefinitivo.png');

end;

if IzquierdaOnly(C,F) then begin

img.Picture.LoadFromFile('images/IzquierdaOnly.png');

sw:=true;

end else if sw=False then begin

img.Picture.LoadFromFile('images/TierraDefinitivo.png');

end;

if(lab[f,c]=1)then begin

img.Picture.LoadFromFile('images/Pastito.png')

end;

img.Left:=f\*dx;

img.Top:=c\*dy;

img.Width:=dx;

img.Height:=dy;

img.Stretch:=true;

sw:=False;

end;

img:=TImage.Create(ctx);

img.Parent:=ctx;

img.Picture.LoadFromFile('images/Ratoncito.png');

img.Left:=raton.col\*dx;

img.Top:=raton.fil\*dy;

img.Width:=dx-20;

img.Height:=dy-20;

img.Stretch:=true;

filAntigua:=2;

ColAntigua:=1;

end;

end;

ctx.Show;

end;

procedure Juego.DibujarRaton(DireccioMirando:integer);

var

f,c,dx,dy,SumaF,SumaC:integer;

img:TImage;

img2:Timage;

sw:boolean;

respuesta:string;

begin

f:=FilAntigua;

c:=ColAntigua;

dx:=80;

dy:=80;

case DireccioMirando of

0 : begin

img:=TImage.Create(ctx);

img.Parent:=ctx;

img.Picture.LoadFromFile('images/RatonMirandoAbajo.png');

img.Left:=raton.col\*dx;

img.Top:=raton.fil\*dy;

img.Width:=dx-20;

img.Height:=dy-20;

img.Stretch:=true;

end;

1 : begin

img:=TImage.Create(ctx);

img.Parent:=ctx;

img.Picture.LoadFromFile('images/RatoncitoArriba.png');

img.Left:=raton.col\*dx;

img.Top:=raton.fil\*dy;

img.Width:=dx-20;

img.Height:=dy-20;

img.Stretch:=true;

end;

2 : begin

img:=TImage.Create(ctx);

img.Parent:=ctx;

img.Picture.LoadFromFile('images/RatoncitoMirandoIzq.png');

img.Left:=raton.col\*dx;

img.Top:=raton.fil\*dy;

img.Width:=dx-20;

img.Height:=dy-20;

img.Stretch:=true;

end;

3 : begin

img:=TImage.Create(ctx);

img.Parent:=ctx;

img.Picture.LoadFromFile('images/Ratoncito.png');

img.Left:=raton.col\*dx;

img.Top:=raton.fil\*dy;

img.Width:=dx-20;

img.Height:=dy-20;

img.Stretch:=true;

end;

end;

img2:=TImage.Create(ctx);

img2.Parent:=ctx;

if EsEsquinaDer(f,c) then begin

img2.Picture.LoadFromFile('images/EsquinaDer.png');

sw:=true;

end else begin

img2.Picture.LoadFromFile('images/TierraDefinitivo.png');

end;

if EsEsquinaIZ(f,c) then begin

img2.Picture.LoadFromFile('images/EsquinaIZ.png');

sw:=true;

end else if sw=False then begin

img2.Picture.LoadFromFile('images/TierraDefinitivo.png');

end;

if Vertical(F,C) then begin

img2.Picture.LoadFromFile('images/VerticalImagen.png');

sw:=true;

end else if sw=False then begin

img2.Picture.LoadFromFile('images/VerticalImagen.png');

end;

if EsEsquinaDerDown(f,c) then begin

img2.Picture.LoadFromFile('images/EsquinaDerDown.png');

sw:=true;

end else if sw=False then begin

img2.Picture.LoadFromFile('images/TierraDefinitivo.png');

end;

if CentroSuperior(F,c) then begin

img2.Picture.LoadFromFile('images/GrassUp.png');

sw:=true;

end else if sw=False then begin

img2.Picture.LoadFromFile('images/TierraDefinitivo.png');

end;

if EsquinaIZUp(f,c) then begin

img2.Picture.LoadFromFile('images/EsquinaIzSup.png');

sw:=true;

end else if sw=False then begin

img2.Picture.LoadFromFile('images/TierraDefinitivo.png');

end;

if IzquierdaOnly(f,c) then begin

img2.Picture.LoadFromFile('images/IzquierdaOnly.png');

sw:=true;

end else if sw=False then begin

img2.Picture.LoadFromFile('images/TierraDefinitivo.png');

end;

if IzquierdaEncerrado(F,C) then begin

img2.Picture.LoadFromFile('images/IzquierdaCerrado.png');

sw:=true;

end else if sw=False then begin

img2.Picture.LoadFromFile('images/TierraDefinitivo.png');

end;

if(lab[c,f]=1)then begin

img2.Picture.LoadFromFile('images/Pastito.png');

end;

img2.Left:=c\*dx;

img2.Top:=f\*dy;

img2.Width:=dx;

img2.Height:=dy;

img2.Stretch:=true;

FilAntigua:=raton.fil;

ColAntigua:=Raton.col;

if(raton.fil=ini.fil)and(raton.col=ini.col)then

ShowMessage('En la Entrada ...!!!');

if(raton.fil=fin.fil)and(raton.col=fin.col)then begin

sndPlaySound('sounds/end.wav', SND\_NODEFAULT Or SND\_ASYNC or SND\_FILENAME);

ShowMessage('Llegaste a la Salida...!!!');

if nivel=3 then begin

Respuesta:=inputbox('¿Desea volver a jugar?','Si/No','');

respuesta:=Uppercase(respuesta);

respuesta:=trim(respuesta);

if respuesta='SI' then begin

nivel:=0;

end;

if respuesta='NO' then begin

ctx.close;

ctx.Hide;

end;

end;

self.CrearLab(nivel+1,2,1);

end;

end;

procedure Juego.MoverRaton(dir:Direccion);

var up,down,rigth,left:boolean;

f,c,suma:integer;

direccionMirada:integer;

begin

//el raton no puede pasar por las paredes

//Mejorar el Codigo

up:=false;

down:=false;

rigth:=false;

left:=false;

case dir of

Arriba:begin

raton.fil:=raton.fil-1;

Up:=true;

direccionMirada:=1;

end;

Abajo :begin

raton.fil:=raton.fil+1;

down:=true;

direccionMirada:=0;

end;

Derecha:begin

raton.col:=raton.col+1;

Rigth:=true;

direccionMirada:=3;

end;

Izquierda:begin

raton.col:=raton.col-1;

left:=true;

direccionMirada:=2;

end;

end;

f:=raton.fil;

c:=raton.col;

suma:=f+c;

if (lab[c,f]=1) or (suma=2) then begin

if up=true then begin

raton.fil:=raton.fil+1;

end;

if down=true then begin

raton.fil:=raton.fil-1;

end;

if rigth=true then begin

raton.col:=raton.col-1;

end;

if left=true then begin

raton.col:=raton.col+1;

end;

end else if lab[c,f]=0 then begin

DibujarRaton(DireccionMirada);

end;

end;

procedure Juego.setnom(n: string);

begin

nom:=n;

end;

function Juego.getnom(): string;

begin

result:=nom

end;

procedure Juego.crear();

begin

if est=0 then begin

assign(fe,nom);

{$I-}

rewrite(fe);

{$I+}

if (IOResult<>0) then begin

showmessage('Error al crear al archivo con tipo');

exit;

end;

est:=1; //Modo escritura

end else begin

showmessage('El archivo con tipo se encuentra abierto');

end;

end;

procedure Juego.abrir();

begin

if est=0 then begin

Assign(fe,nom);

{$I-}

reset(fe);

{$I+}

if (IOresult<>0) then begin

showmessage('Error al abrir al archivo con tipo');

exit;

end;

est:=2; //Modo lectura

end else begin

showmessage('El archivo con tipo se encuentra abierto');

end;

end;

procedure Juego.escribir(reg: posicion);

begin

if est<>0 then begin

write (fe,reg);

end else begin

showmessage('Error al escribir, el archivo con tipo se encuentra cerrado');

end;

end;

procedure Juego.leer(var reg: posicion);

begin

if est<>0 then begin

read (fe,reg);

end else begin

showmessage('Error al leer, el archivo con tipo se encuentra cerrado');

end;

end;

procedure Juego.cerrar;

begin

close(fe);

est:=0;

end;

function Juego.FinDeArchivo(): boolean;

begin

result:=eof(fe);

end;

procedure Juego.cargar();

var reg:posicion;

begin

self.crear;

reg.fil:=raton.fil;

reg.col:=raton.col;

reg.nivelActual:=nivel;

self.escribir(reg);

self.cerrar;

end;

function Juego.GuardarPosF(): integer;

begin

result:=raton.fil;

end;

function Juego.GuardarPosC(): integer;

begin

result:=raton.col;

end;

procedure Juego.Posicionar(pos: integer);

begin

If (pos>=0) and (pos<=filesize(fe)) then begin

seek(fe,pos);

end else begin

showmessage('Posicion fuera de rango');

end;

end;

function Juego.EsEsquinaDer(F, C: integer): boolean;

begin

if nivel=1 then begin

case f of

2 : begin

if c=2 then begin

result:=true;

end;

end;

3: begin

If c=3 then begin

result:=true;

end;

end;

end;

end;

if nivel=2 then begin

case f of

2 : begin

if c=2 then begin

result:=true;

end;

end;

end;

end;

if nivel=3 then begin

case f of

2 : begin

if c=3 then begin

result:=true;

end;

if c=7 then begin

result:=true;

end;

end;

7: begin

if c=3 then begin

result:=true;

end;

end;

end;

end;

end;

function Juego.EsEsquinaIz(F, c: integer): boolean;

begin

if nivel=1 then begin

case f of

3 : begin

if c=2 then begin

result:=true;

end;

end;

end;

end;

if nivel=2 then begin

case f of

5 : begin

if c=2 then begin

result:=true;

end;

end;

end;

end;

if nivel=3 then begin

case f of

7 : begin

if c=2 then begin

result:=true;

end;

end;

end;

end;

end;

function Juego.Vertical(f, c:integer): boolean;

begin

if nivel=1 then begin

case f of

4 : begin

if c=3 then begin

result:=true;

end;

end;

end;

end;

if nivel=2 then begin

case f of

3 : begin

if c=2 then begin

result:=true;

end;

if c=5 then begin

result:=true;

end;

end;

4 : begin

if c=2 then begin

result:=true;

end;

if c=5 then begin

result:=true;

end;

end;

end;

end;

if nivel=3 then begin

case f of

3 : begin

if c=2 then begin

result:=true;

end;

if c=5 then begin

result:=true;

end;

if c=7 then begin

result:=true;

end;

end;

4 : begin

if c=2 then begin

result:=true;

end;

if c=5 then begin

result:=true;

end;

if c=7 then begin

result:=true;

end;

end;

5 : begin

if c=7 then begin

result:=true;

end;

end;

6 : begin

if c=2 then begin

result:=true;

end;

if c=7 then begin

result:=true;

end;

end;

8 : begin

if c=5 then begin

result:=true;

end;

end;

end;

end;

end;

function Juego.EsEsquinaDerDown(f, c: integer): boolean;

begin

if nivel=2 then begin

case f of

5 : begin

if c=5 then begin

result:=true;

end;

end;

end;

end;

if nivel=3 then begin

case f of

5 : begin

if c=5 then begin

result:=true;

end;

end;

7 : begin

if c=7 then begin

result:=true;

end;

end;

end;

end;

end;

function Juego.CentroSuperior(f, c: integer): boolean;

begin

if nivel=2 then begin

case f of

2 : begin

if c=5 then begin

result:=true;

end;

end;

end;

end;

if nivel=3 then begin

case f of

2 : begin

if c=2 then begin

result:=true;

end;

end;

end;

end;

end;

function Juego.EsquinaIzUp(f, c:integer): boolean;

begin

if nivel=3 then begin

case f of

2 : begin

if c=5 then begin

result:=true;

end;

end;

7 : begin

if c=5 then begin

result:=true;

end;

end;

end;

end;

end;

function Juego.IzquierdaOnly(f, c: integer): boolean;

begin

if nivel=3 then begin

case f of

5 : begin

if c=2 then begin

result:=true;

end;

end;

end;

end;

end;

function Juego.IzquierdaEncerrado(f, c: integer): boolean;

begin

if nivel=2 then begin

case f of

2 : begin

if c=4 then begin

result:=true;

end;

end;

end;

end;

end;

function Juego.GetNivel(): integer;

begin

result:=nivel;

end;

end.