Universidad Rafael Landívar

Facultad de Ingeniería

Ingeniería en Informática y Sistemas

Introducción a la programación

Catedrática: inga. Karen Liska

**Proyecto “Snake”**

Cortez Juárez, Génesis Dayana E. - 1059618

Gálvez Arriaza, Andres Sebastian - 1024718

Guatemala, 10 de marzo de 2018

**Introducción**

La clase de “Introducción a la programación” es aquella que cada estudiante de ingeniería ha de llevar durante el primer año de sus estudios. Es la encargada de introducir a los alumnos al mundo de la programación, brindando conocimientos básicos para el correcto uso de lenguajes de programación básicos, la manera correcta de elaborar un programa, las bases que cada lenguaje o programa requieren.

Durante el transcurso de la clase se va adquiriendo conocimiento sobre algún programa en específico, pero a su vez se desarrolla la lógica del alumno para que de esta manera pueda ser capaz de resolver problemas básicos haciendo uso de las herramientas brindadas. Para fortalecer dicho aprendizaje, se realizan en cada ciclo dos “proyectos”. Dichos casos son especialmente diseñados para que el alumno pueda emplear sus conocimientos y desarrollar nuevos. Dependiendo del nivel de los estudiantes se les brinda los reglamentos que deben seguir para la elaboración del programa solicitado.

En el primer ciclo del año 2018, se ha solicitado la elaboración de un programa similar el juego ya conocido como “Snake”. Dicho juego fue lanzado a mediados de los 70, siendo en el ahora un juego con una gran popularidad, la cual obtuvo al momento que fue integrada como juego oficial en los celulares Nokia. Para la elaboración de dicho juego se dejaron requerimientos obligatorios, limitaciones y se obviaron ciertas funciones del juego.

El juego “Snake” es aquel en el cual el usuario controlaba una pequeña serpiente la cual no podía tocar los límites de la pantalla, debía comer cada fruta que aparecía en la pantalla y mientras lo hacía iba creciendo. A su vez que crecía se recalcaba que no podía colisionar con su cola ya que esto, al igual que con los límites, causaría el fin del juego. Para el proyecto, se solicitó que la serpiente fuera controlada al igual que en el juego original, por el usuario y siguiendo las mismas limitaciones, los cuales eran, no colisionar con los límites o su cola, además de no poder retroceder sobre sí misma. Las diferencias serían, la serpiente no debe comer ninguna fruta durante el juego, por tal motivo, esta no crecería. Además, la serpiente se ha de mover uno a uno, en otras palabras, no tendría un movimiento propio a menos que el usuario lo solicite. El tamaño de la pantalla sería solicitado al principio del juego por el programa, así el usuario podría decidir sus límites. Por último, se le deberá mostrar al usuario las coordenadas exactas de la cabeza y cola de la serpiente.

**Objetivos**

* **Objetivo General**
  + Hacer uso correcto de los conocimientos adquiridos durante la clase además del desarrollo de la lógica para la realización del proyecto.
* **Objetivos Específicos**
  + Crear un programa similar al juego “snake” con diferencias ya establecidas.
  + Plantear de manera correcta el procedimiento a realizar para el proyecto.

**Análisis del Proyecto**

**Entradas:** Alto y largo de la pantalla. Direcciones ingresadas por las flechas del teclado.

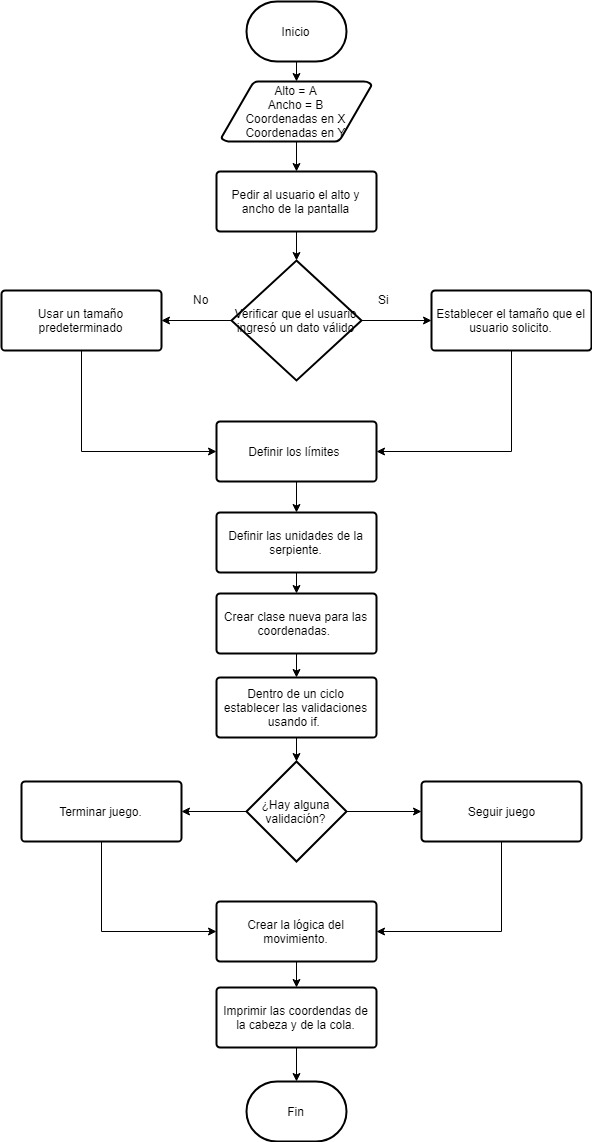
**Procesos:** Impresión de coordenadas, localización de la serpiente, posibles colisiones, inicio del juego, fin del juego.

**Salidas:** Coordenadas de la cabeza, coordenadas de la cola.

**Validaciones:** No puede retroceder sobre si misma. No es posible sobre pasar los límites de la pantalla. No puede pasar sobre si misma o tocar parte de su cuerpo. No debe ir creciendo. Solo se puede mover si el usuario lo desea.

**Diseño del Proyecto**

**Diagrama de flujo:**



**Pseudocódigo:**

Class\_Snake

Begin

int ancho, alto, limiteX, limiteY;

Write("Bienvenido al juego Snake");

Write("Por favor ingresar ancho de la pantalla");

if (int.TryParse(Read(), out ancho) == false)

{

Write("Usted no ingreso un dato valido, se usara el predefinido");

ancho = 10;

}

Write("Por favor ingresar alto de la pantalla");

if (int.TryParse(Read(), out alto) == false)

{

Write(“Usted no ingreso un dato valido, se usara el predefinido");

alto = 10;

}

limiteX = ancho / 2;

limiteY = alto / 2;

coordenada cabeza = new coordenada(0, 0);

coordenada parte1 = new coordenada(-1, 0);

coordenada parte2 = new coordenada(-2, 0);

coordenada cola = new coordenada(-3, 0);

coordenada nuevo = new coordenada(0,0);

bool seguir = true;

do

{

var key = ReadKey(true).Key;

switch (key)

{

case Key.DownArrow:

nuevo = new coordenada(cabeza.X, cabeza.Y - 1);

break;

case Key.LeftArrow:

nuevo = new coordenada(cabeza.X - 1, cabeza.Y);

break;

case Key.RightArrow:

nuevo = new coordenada(cabeza.X + 1, cabeza.Y);

break;

case Key.UpArrow:

nuevo = new coordenada(cabeza.X, cabeza.Y + 1);

break;

}

if ((nuevo.X == limiteX) || (nuevo.Y == limiteY) || (nuevo.X == -limiteX) || (nuevo.Y == -limiteY))

{

Clear();

Write("Fin del juego, toco los limites");

seguir = false;

}

else if ((nuevo.X == parte1.X) && (nuevo.Y==parte1.Y))

{

}

else if ((nuevo.X == cola.X) && (nuevo.Y == cola.Y))

{

Clear();

Write("Fin del juego, por choque con cola");

seguir = false;

}

else

{

cola = parte2;

parte2 = parte1;

parte1 = cabeza;

cabeza = nuevo;

}

Write("La coordenada de la cabeza en x es " + cabeza.X.ToString() + " La coordenada de la cabeza en Y es " + cabeza.Y.ToString());

Write("La coordenada de la cola en x es " + cola.X.ToString() + " La coordenada de la cola en Y es " + cola.Y.ToString());

} while (seguir == true);

Write("Gracias por jugar");

End\_Snake

**Diagrama de clases**

|  |
| --- |
| **Class Program** |
| + int ancho  + int alto  + int limiteX  + int limiteY  + bool seguir  + var key  + Coordenada cabeza  + Coordenada parte1  + Coordenada parte2  + Coordenada parte3  + Coordenada cola  + Coordenada nuevo |
| + cabeza ()  + parte1 ()  + parte2 ()  + parte3 ()  + cola ()  + nuevo () |

|  |
| --- |
| **Class Coordenada** |
| + int x  + int y  + Coordenada X  + Coordenada Y |
| + Coordenada () |

**Conclusiones**

1. Para realizar un proyecto de informática, es sumamente indispensable realizar un análisis previo para minimizar costos y tiempo de desarrollo.
2. Los métodos son funcionalidades, operaciones o servicios que describen el comportamiento de un objeto.
3. Para facilitar la codificación y evitar procesos muy extensos, es conveniente hacer uso de estructuras repetitivas, que, como su nombre indica, repiten una secuencia de instrucciones un número determinado de veces.

**E-Grafía**

1. GioCode (27 de agosto de 2010). Como hacer un juego de snake en C# (Windows Forms) [Video file]. Recuperado de <https://www.youtube.com/watch?v=sZ9Vtcokkb4>
2. Petedk Codes (21 de septiembre de 2016). Snake Game Part 1 [Video file]. Recuperado de https://www.youtube.com/watch?v=TGNBBdgYcEs
3. Hào Phan Phú (11 de julio de 2017). [C# console game] Snake [Video file]. Recuperado de https://www.youtube.com/watch?v=b-SvyCHHWsQ