

Máquina Tragaperras - Proyecto en C#



Grupo: Mario Martínez Lozano

Grupo: C

Curso: 2023/2024

Asignatura: Programación Orientada a Objetos

ÍNDICE

1. Introducción.....	Página 3
2. Descripción del Proyecto.....	Página 4-6
3. Desafíos del Proyecto.....	Página 7
4. Conclusiones.....	Página 8
5. Bibliografía.....	Página 9

INTRODUCCIÓN

Mario Martínez Lozano

Esta es una aplicación de consola que simula una máquina tragaperras. El programa muestra un menú principal con las opciones: Jugar, Mostrar premios, Cargar premios (opción exclusiva del Admin) y Salir.

- Jugar: muestra las posiciones iniciales aleatorias de los símbolos en tres rodillos (3 x 3) y ofrece dos opciones: Jugar (hace girar los rodillos de nuevo) y Salir (vuelve al menú principal).

Si el jugador obtiene tres números iguales en la línea central, se muestra una frase motivacional.

- Mostrar premios: lista todos los premios disponibles y solicita el ID del premio para mostrar detalles específicos, incluyendo el nombre, la línea de símbolos y la frase asociada.
- Cargar premios (Admin): solicita el nombre del archivo CSV, verifica su existencia y reemplaza los premios actuales por los del archivo cargado. Esta opción requiere una contraseña.
- Salir: finaliza la ejecución del programa (sin utilizar `Environment.Exit(0)`).

El programa utiliza generadores de números aleatorios mediante la función `Random` que permite mediante una matriz simular esta máquina tragaperras.

Los premios se dividen en dos categorías: simples y aleatorios. Los premios simples tienen un consejo específico, mientras que los aleatorios tienen dos consejos con una probabilidad definida en el CSV.

El archivo CSV para cargar premios utiliza ";" como separador y contiene campos para el tipo de premio, nombre del premio, símbolos de los rodillos y consejos asociados a cada premio.

DESCRIPCIÓN DEL PROYECTO

Esta es una aplicación de consola que simula una máquina tragaperras. El programa muestra un menú principal con las opciones: Jugar, Mostrar premios, Cargar premios (opción exclusiva del Admin) y Salir.

Funcionamiento General del Slot Machine:

6. Jugar:

- Al seleccionar esta opción, el programa muestra las posiciones iniciales aleatorias de los símbolos en tres rodillos, dispuestos en una matriz de 3x3.
- Los rodillos giran utilizando un generador de números aleatorios para asegurar resultados justos y equitativos.
- El jugador tiene dos opciones:
 - **Jugar:** Hace girar los rodillos nuevamente, reordenando aleatoriamente los símbolos.
 - **Salir:** Vuelve al menú principal sin realizar ninguna acción adicional.
- Si el jugador obtiene tres símbolos iguales en la línea central (la fila del medio de la matriz 3x3), se muestra una frase motivacional en lugar de premios monetarios.

7. Mostrar premios:

- Al seleccionar esta opción, el programa lista todos los premios disponibles en la máquina tragaperras.
- Luego de listar los premios, se solicita al usuario que ingrese el identificador (ID) del premio para mostrar detalles específicos, que incluyen:
 - Nombre del premio.
 - La combinación de símbolos necesarios.
 - La frase de consejo asociada al premio.

8. Cargar premios (Admin):

- Esta opción está restringida a los usuarios y requiere la introducción de una contraseña para acceder.
- El proceso de carga de premios sigue los siguientes pasos:
 - Se solicita al administrador el nombre del archivo CSV desde el cual se cargarán los premios.
 - El programa verifica la existencia del archivo.
 - Si el archivo existe y es válido, el programa reemplaza los premios actuales en la máquina tragaperras con los premios especificados en el archivo cargado.
- El archivo CSV debe utilizar el punto y coma (;) como separador y contener los siguientes campos:
 - Tipo de premio (1 para premios simples, 2 para premios aleatorios).
 - Nombre del premio.
 - Símbolos requeridos para el primer, segundo y tercer rodillo.
 - Consejo específico (para premios simples) o dos consejos (para premios aleatorios con su porcentaje de probabilidad).
 - Probabilidad para el primer consejo (solo para premios aleatorios).

9. Salir:

- Esta opción finaliza la ejecución del programa de manera ordenada, asegurando que todos los recursos se liberen adecuadamente y sin utilizar `Environment.Exit(0)`.

Funcionamiento de los rodillos:

El programa utiliza generadores de números aleatorios para determinar las posiciones de los símbolos en los rodillos. Esto garantiza que los resultados sean justos y aleatorios (utilizando la función `Random` de Visual Studio).

Funcionamiento de los premios:

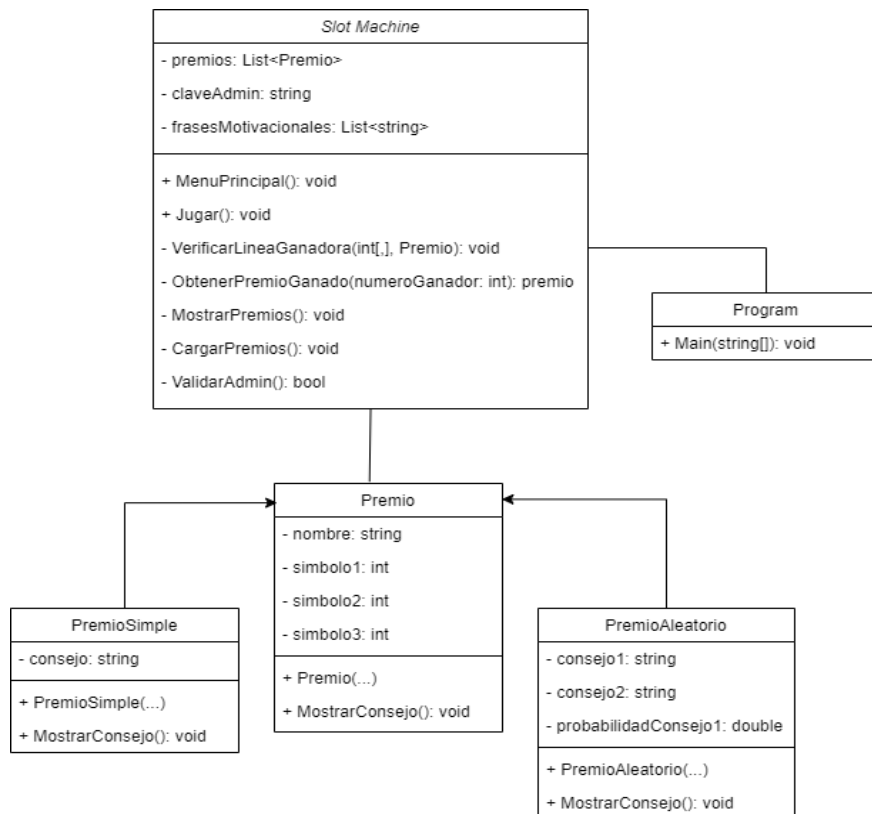
Los premios en la máquina tragaperras se dividen en dos categorías principales:

10. Premios simples:

- Estos premios están asociados con un consejo específico que se muestra cuando el jugador obtiene la combinación ganadora de símbolos.
- Cada premio simple tiene los siguientes atributos:
 - Nombre del premio.
 - Símbolos requeridos en el primer, segundo y tercer rodillo.
 - Consejo específico mostrado al ganar el premio.

11. Premios aleatorios:

- Estos premios presentan dos consejos potenciales con una probabilidad definida para cada uno.
- Cada premio aleatorio tiene los siguientes atributos:
 - Nombre del premio.
 - Símbolos requeridos en el primer, segundo y tercer rodillo.
 - Dos consejos potenciales.
 - Probabilidad para el primer consejo, donde la probabilidad para el segundo consejo es 1 menos la probabilidad del primer consejo.



DESAFIOS DEL PROYECTO

Uno de los aspectos más complicados ha sido implementar el sistema de premios aleatorios. Utilizar la clase Random para asegurarme de que, según la probabilidad definida en el archivo CSV, se muestre uno u otro consejo al ganar un premio.

```

// METODO OVERRIDE MUESTRA CONSEJOS ALEATORIOS (sobrescribe al metodo abstracto de la clase padre premio)
1 referencia
public override void MostrarConsejo()
{
    // SEGUN LA PROBABILIDAD SALE EL CONSEJO
    Random rand = new Random();
    double randomConsejo = rand.NextDouble();
    if (randomConsejo <= ProbabilidadConsejo1)
    {
        Console.WriteLine($" ¡Felicidades! Has ganado un {Nombre} y un Consejo: {Consejo1}");
    }
    else
    {
        Console.WriteLine($" ¡Felicidades! Has ganado un {Nombre} y un Consejo: {Consejo2}");
    }
}
  
```

Otra dificultad fue la parte del código que verifica la línea ganadora y asocia el premio correspondiente con los símbolos de los rodillos de la línea central. Hubo varios casos en los que el premio no se asignaba ya que no estaba comparando bien la fila central de la matriz con la lista de los productos cargados.

```
// BUSCAR PREMIO CORRESPONDIENTE A LA LINEA GANADORA
1 referencia
private Premio ObtenerPremioGanado(int numeroGanador)
{
    // BUSCAR EN LA LISTA PREMIOS LOS SIMBOLOS CORRESPONDIENTES
    foreach (var premio in premios)
    {
        if (premio.Simbolo1 == numeroGanador && premio.Simbolo2 == numeroGanador && premio.Simbolo3 == numeroGanador)
        {
            return premio;
        }
    }
    return null;
}
```

Además, otro problema fue el asignar correctamente los campos, ya que los consejos 1 y 2 y el valor aleatorio no lo tienen los premios simples a diferencia de los aleatorios que si los tienen.

```
if (tipoPremio == 1) // PREMIO SIMPLE
{
    string consejo = campos[5]; // CONSEJO ÚNICO
    PremioSimple premioSimple = new PremioSimple(nombrePremio, simbolo1, simbolo2, simbolo3, consejo);
    nuevosPremios.Add(premioSimple);
}
else if (tipoPremio == 2) // PREMIO ALEATORIO
{
    string consejo1 = campos[5]; // CONSEJO ALEATORIO 1
    string consejo2 = campos[6]; // CONSEJO ALEATORIO 2
    double probabilidad = double.Parse(campos[7]); // PROBABILIDAD DEL CONSEJO ALEATORIO 1

    PremioAleatorio premioAleatorio = new PremioAleatorio(nombrePremio, simbolo1, simbolo2, simbolo3, consejo1, consejo2, probabilidad);
    nuevosPremios.Add(premioAleatorio);
}
```

CONCLUSIONES

Durante el desarrollo de este proyecto, he aprendido a utilizar mejor las funciones Random para manejar probabilidades y números aleatorios.

También he mejorado en la carga y lectura de archivos de texto CSV, lo cual fue importante para gestionar los premios y sus consejos según sus símbolos y su tipo (simple / aleatorio).

En cuanto al manejo de matrices, he aprendido a manejarlas mejor, importante para generar los números aleatorio o para verificar la línea ganadora, pero solo la central, es decir, cuando los símbolos de la central fueran iguales y así mostrar una frase motivacional con su consejo correspondiente al premio ganado según sus símbolos.

En general, este proyecto me ha hecho mejorar en conceptos de programación orientada a objetos como la agregación, la herencia, las excepciones y el polimorfismo.

La mayor parte del aprendizaje ha sido a base de prueba y error modificando constantemente el código para que no salieran errores, pero sobre todo para que funcionara de la forma correcta como se indicaba en la tarea.



BIBLIOGRAFÍA

- BillWagner. (s. f.). *C# Guide - .NET managed language*. Microsoft Learn. <https://learn.microsoft.com/en-us/dotnet/csharp/>
- *Stack Overflow - where developers learn, share, & build careers*. (s. f.). Stack Overflow. <https://stackoverflow.com/>
- *Reddit. (n.d.). r/csharp - A subreddit for c# programming language*. Reddit. Retrieved June 14, 2024, from. <https://www.reddit.com/r/csharp/>
- *Apuntes y ejercicios de Canvas – Programación Orientada a Objetos*.

