

Universidad Rafael Landívar

Facultad de Ingeniería

Ingeniería Química

Lab. de introducción a la programación

Catedrático: Ing. Edwin Chocoy

PROYECTO PRÁCTICO NO. 02

DADOS

Bárbara Castellanos

1049923

Ciudad de Guatemala, 13 de noviembre de 2023

ÍNDICE

INTRODUCCIÓN.....	1
ANÁLISIS	2
ENTRADAS.....	2
SALIDAS.....	2
PROCESOS.....	3
RESTRICCIONES.....	3
DISEÑO	4
DIAGRAMA DE FLUJO	4
CONCLUSIONES	5
RECOMENDACIONES.....	5
REFERENCIAS	6
BIBLIOGRÁFICAS	6
ELECTRÓNICAS	6
ANEXOS.....	7
MANUAL DE USUARIO	7

INTRODUCCIÓN

En este documento, se presenta el desarrollo de un programa de juego de dados en el lenguaje de programación C#. El objetivo principal es simular de manera realista el desarrollo de un juego de dados, donde los usuarios pueden participar, ingresando la cantidad de partidas y tiros por partida que desean realizar.

El código desarrollado permite a los usuarios interactuar con el programa, ingresando información esencial como la cantidad de partidas y tiros deseados. Se incorpora la generación aleatoria de números mediante la clase Random para simular los resultados de lanzar dados en cada tiro.

El documento se estructura en secciones que exploran las entradas, salidas, procesos y restricciones presentes en el código, proporcionando una visión completa de su funcionamiento. Además, se brindan conclusiones y recomendaciones detalladas para orientar futuras mejoras y optimizaciones en el código del juego de dados.

ANÁLISIS

ENTRADAS

TABLA NO. 01: ENTRADAS DEL CÓDIGO

ENTRADA	ANÁLISIS
Ingreso de la cantidad de partidas “partidas”	El usuario ingresa la cantidad de partidas que desea jugar, en donde debe ser un número entero y positivo.
Ingreso de tiros por partida “tiros”	El usuario debe ingresar la cantidad de tiros a realizar por cada partida indicada, en donde debe ser un número entero y positivo.

Fuente: Elaboración propia, (2023).

SALIDAS

TABLA NO. 02: SALIDAS DEL CÓDIGO

SALIDAS	ANÁLISIS
Salida de la presentación de resultados	El programa muestra un mensaje en donde se indica que se mostrará el resultado final del juego realizado.
Salida de el ganador de la partida	El programa muestra el ganador de la partida estableciendo una condición en la cual, si las partidas ganadas del jugador son mayores que la mitad de las partidas jugadas, el jugador es el ganador, de lo contrario, la Casa gana.
Salida de las tiradas realizadas por cada partida	El programa muestra el total de los tiros realizados por cada partida llamando a la variable “tiros”.
Salida de las partidas ganadas por el jugador	El programa muestra las partidas ganadas por el jugador, quiere decir los tiros en los cuáles el jugador adquirió puntos.
Salida de la probabilidad de ganar	Después de calcular la probabilidad de ganar del jugador el cuál se hizo mediante una regla de tres en la que se multiplicó las tiradas ganadas por el jugador por 100 y luego se dividió por la cantidad de partidas, se muestra la probabilidad en porcentaje.
Salida de los tiros con números pares y números impares	El programa muestra los tiros con suma par, quiere decir que toma en cuenta la suma de los dados en cada tiro y considera si es par e impar.
Salida de tiros con números iguales	El programa muestra el tiro que contiene los números iguales resultantes de los dos dados.

Puntaje final del jugador y de la Casa	Al finalizar el programa, se muestra el puntaje total del jugado y de la Casa por todas las partidas jugadas.
--	---

Fuente: Elaboración propia, (2023).

PROCESOS

TABLA NO. 03: PROCESOS DEL CÓDIGO

PROCESOS	ANÁLISIS
Generación de números aleatorios “Random”	En este proceso se utilizan 2 dados para generar números aleatorios del 1 al 6 (incluyendo ambos) en cada tiro de las partidas.
Puntajes	Se van acumulando los puntajes para el jugador y la casa en cada tiro, en donde se va sumando el puntaje final al acabar cada partida.
Conteo de estadísticas	Se cuentan los tiros pares, impares e iguales de cada partida, además se mantiene un registro de las partidas ganadas por el jugador.
Cálculo de Probabilidad	Se calcula la probabilidad de que el jugador gane en donde se expresa en porcentaje.

Fuente: Elaboración propia, (2023).

RESTRICCIONES

TABLA NO. 04: RESTRICCIONES DEL CÓDIGO

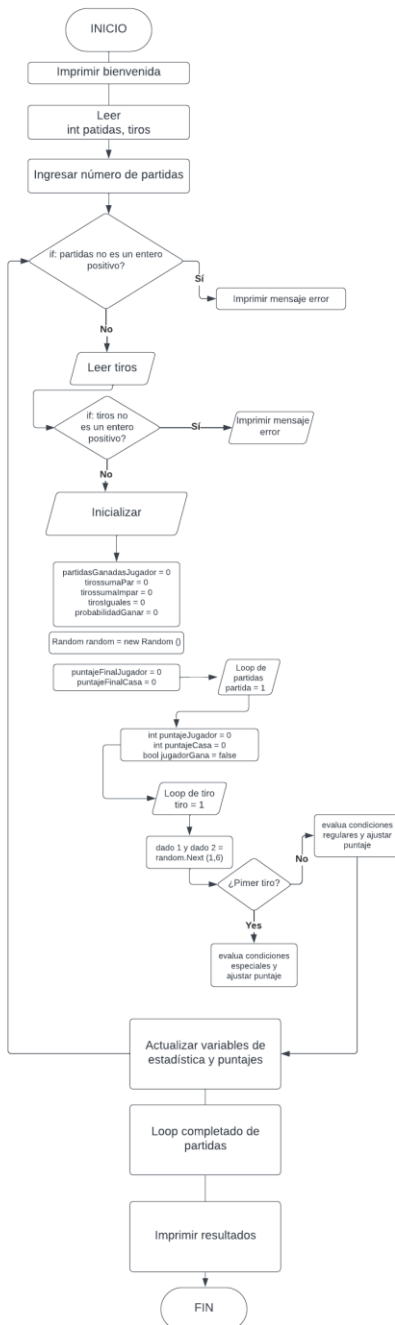
RESTRICCIONES	ANÁLISIS
Validación de entradas	La cantidad de partidas y tiros debe ser un número entero y positivo, si no se cumple esta condición, el programa le solicita al usuario que ingrese los valores nuevamente.
Condiciones especiales	Se tienen condiciones especiales en el primer tiro que afectan el puntaje del jugador y la Casa.
Espera simulada “Thread.Sleep(1000)”	Se utiliza esta función para simular el tiempo estimado de 1 segundo que se toma entre cada tiro, más que todo para convertirlo más realista.
Cálculo de probabilidad	La probabilidad de ganar se calcula correctamente si la cantidad de partidas ingresadas por el usuario es mayor que cero.

Fuente: Elaboración propia, (2023)

DISEÑO

DIAGRAMA DE FLUJO

DIAGRAMA NO. 01: DIAGRAMA DE FLUJO DEL CÓDIGO REALIZADO



Fuente: Elaboración propia en el programa de Lucidchart, (2023).

DIAGRAMA NO. 02: DIAGRAMA DE CLASES DEL CÓDIGO REALIZADO

CLASE
<i>Program</i>
Main (string[] args)
InicializarJuego()
JugarPartida(int partidas, int tiros)
Calcular Probabilidad ()
Mostrar resultados ()
<i>Random</i>
Random random = new Random()

Fuente: Elaboración propia en el programa de Microsoft Excel, (2023).

CONCLUSIONES

1. Se creó un programa con lenguaje C# que gestiona la información sobre las rutas proporcionadas en el documento brindado con las instrucciones del proyecto, en el cual el código generado permite al usuario ingresar la cantidad de partidas y tiros dentro de las mismas para generar un juego basado en el azar.
2. El programa se comunica con el usuario a través de la consola por medio de un texto en donde el código va generando la información detallada de cada partida y los tiros efectuados en la misma, en donde esta información se irá guardando en las variables correspondientes para generar las respuestas al final del juego.
3. La implementación de la clase Random en el código demuestra ser esencial para la simulación realista del juego de dados. La utilización de esta clase permitió el poder generar números aleatorios, específicamente del 1 al 6 para simular los resultados de las tiradas de los dados.

RECOMENDACIONES

1. Realizar pruebas exhaustivas para asegurar el correcto funcionamiento del código en diferentes escenarios, especialmente alrededor de condiciones que se puedan presentar.
2. Asegurarse que el código siga un orden lógico y así poder obtener los datos requeridos de forma ordenada y coherente.
3. Proporcionar una documentación más detallada sobre las reglas del juego y el significado de los cálculos efectuados para facilitar su comprensión.

4. Declarar la instancia "Random" fuera del bucle principal, evitando así la creación repetitiva de instancias o valores que podrían afectar la aleatoriedad.

REFERENCIAS

BIBLIOGRÁFICAS

N/A

ELECTRÓNICAS

- Dotnet-Bot. (s. f.-b). Thread.Sleep método (System.Threading). Microsoft Learn. <https://learn.microsoft.com/es-es/dotnet/api/system.threading.thread.sleep?view=net-7.0>

FUNCIÓN: Utilizado para lograr que el programa tarde un tiempo en ejecutar el siguiente tiro. A través de esto, se logró crear el efecto de la espera entre tiros de los dados.

- BillWagner. (2023, mayo 10). Palabra clave out (modificador genérico): referencia de C# - c#. Microsoft Learn. <https://learn.microsoft.com/es-es/dotnet/csharp/language-reference/keywords/out-generic-modifier>

FUNCIÓN: Utilizado para permitir que el método devuelva múltiples valores a través de los parámetros que se pasan como argumentos.

- Dotnet-Bot. (s. f.-a). Random Constructor (System). Microsoft Learn. <https://learn.microsoft.com/es-es/dotnet/api/system.random.-ctor?view=net-7.0#code-try-1>

FUNCIÓN: Se utilizó para poder completar la simulación del lanzamiento de dados dentro del juego, quiere decir, que este ayudó a generar números aleatorios para los valores de los dados, siendo específicos, números del 1 al 6.

ANEXOS

MANUAL DE USUARIO

A continuación, se presentan las siguientes instrucciones las cuáles el usuario deberá seguir para poder aprovechar al máximo la siguiente herramienta proporcionada de programación:

Bienvenido al juego de dados, ¿Tienes tanta suerte como crees? ¡Ven y ponte a prueba!

Este juego consiste en el lanzamiento aleatorio de dos dados y sumar sus valores, el jugador que sume más puntos al final de la partida, gana. En este caso, será un juego entre “La Casa” y “El jugador”, en el cual la casa vendría siendo el programa y el jugador vendría siendo el usuario.

- Primero, para iniciar el juego, debes ingresar las partidas que desees jugar.

```
Bienvenido al juego de dados, ¿Tienes tanta suerte como crees? ¡Ven y ponte a prueba!  
Ingresa la cantidad de partidas a jugar que desee: 2|
```

- Luego, ingresar el número de tiros que desees jugar por cada partida.

```
Bienvenido al juego de dados, ¿Tienes tanta suerte como crees? ¡Ven y ponte a prueba!  
Ingresa la cantidad de partidas a jugar que desee: 2  
Ingresa la cantidad de tiros por partida a jugar que desee: 2|
```

- El juego comenzará a lo que mostrará en pantalla la siguiente información:

- El número de la partida actual

```
Partida:1
```

- El valor del dado número uno

```
Tiro: 1  
Dado 1: 1 ,
```

- El valor del dado número dos

```
Dado 2: 5
```

- La suma de los valores de los dados

```
Tiro: 1  
Dado 1: 1 , Dado 2: 5  
Suma de los dados lanzados: 6
```

- Al terminar las partidas indicadas se mostrarán los resultados generales del juego, eso quiere decir que se mostrará:
 - El ganador de la partida
 - El número de tiros realizados en cada partida
 - El número de partidas ganadas por el jugador
 - La probabilidad de ganar
 - El número de tiros con suma par
 - El número de tiros con suma impar

- El número de tiros con números iguales
- El puntaje final del jugador
- El puntaje final de la Casa

EJEMPLO:

```
Resultado final:  
El ganador de la partida:La Casa  
Tiros realizados en cada partida:2  
Partidas ganadas por el jugador:1  
Probabilidad de ganar:50%  
Tiros con suma par :4  
Tiros con suma impar:0  
Tiros con números iguales:1  
Puntaje final del jugador:26  
Puntaje final de la Casa:4
```

REGLAS DEL PUNTAJE:

- ✓ Si la suma de los dados es 12 o 6 en el primer tiro, el jugador gana 12 puntos.
- ✓ Si la suma es 4 o 10 en el primer tiro, el jugador pierde y la "Casa" gana 12 puntos.
- ✓ Si la suma es 2, 3, 5, 7, 8 o 9 en el tiro, la suma es el puntaje del jugador o la "Casa".
- ✓ Si la suma de los dados es 11, la "Casa" gana 6 puntos.