

Proyecto práctico No. 2

“Dados”

Guajardo Jiménez, Mia Lucía

Carné: 2002423

Guatemala, 13 de noviembre de 2023**ÍNDICE**

p.

[1. INTRODUCCIÓN 1](#_Toc150699283)

[2. ANÁLISIS 2](#_Toc150699284)

[2.1 ESTRUCTURA: 2](#_Toc150699285)

[2.1.1 RESTRICCIONES 2](#_Toc150699286)

[2.1.2 ENTRADAS 2](#_Toc150699287)

[2.1.3 SALIDAS 2](#_Toc150699288)

[2.1.4 PROBABILIDAD 2](#_Toc150699289)

[3. DISEÑO: 3](#_Toc150699290)

[3.1 DIAGRAMA 3](#_Toc150699291)

[4. CONCLUSIONES 4](#_Toc150699292)

[5. RECOMENDACIONES 5](#_Toc150699293)

# INTRODUCCIÓN

El objetivo del presente proyecto era crear un programa por medio del cual se pudiera simular un juego de dados de dos jugadores. Se jugaría la casa contra el jugador. Se aplicaron las condiciones detalladas en el manual de elaboración para el mismo, las cuales era:

* 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 de la primera tirada 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 punteo del jugador o la “Casa”.
* Un jugador puede perder si la suma tira un 11 antes de haber ganado ningún punto, para este caso la “Casa” gana 6 puntos.

El número de tiros y partidas se le solicitaría al usuario. En este programa se utilizó principalmente los principios de vectores, el ciclo for para utilizar estos mismos, el ciclo while como condicional, y la condicional if.

Adicionalmente de la programación como parte de los resultados se elaboró el cálculo de las probabilidades de ganar para el jugador. En donde se aplicó el principio de probabilidad de combinaciones.

En los resultados se presentaron:

* El ganador de la partida: El jugador o la casa.
* Los tiros realizados en cada partida, cada dado por separado.
* Cuántos tiros ganó el jugador, es decir en cuantos tiros el ganó puntos.
* Cuál era su probabilidad de ganar.
* En cuántos tiros le salieron números pares e impares.
* En cuántos tiros le salieron números iguales.
* Punteo final.

# ANÁLISIS

## ESTRUCTURA:

### RESTRICCIONES

Dentro de las restricciones incluidas en el programa, primero se incluyeron las condiciones del juego detalladas en el manual:

* 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 de la primera tirada 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 punteo del jugador o la “Casa”.
* Un jugador puede perder si la suma tira un 11 antes de haber ganado ningún punto, para este caso la “Casa” gana 6 puntos.

Posteriormente se incluyeron las condiciones de las variables ingresadas. Especialmente el de los tiros de cada partida, ya que se necesitaba un numero par.

* Ingresar un numero par de tiros totales en cada partida.

Se realizaron arreglos dentro de la programación ya que por facilidad del interfaz del usuario y para no causar confusiones, se realizó un control de información mostrada al terminar cada partida.

* Limpiar la consola al terminar cada partida.

### ENTRADAS

Para la funcionalidad del programa se declararon las siguientes variables:

* int sumajuga = 0; Esta variable se utilizó para sumar los dados en cada tiro del jugador.
* int sumacasa = 0; Esta variable se utilizó para sumar los dados en cada tiro del jugador.
* int puntoscasa = 0; Esta variable se utilizó para llevar una cuenta de los puntos de la casa en cada partida.
* int puntosjuga = 0; Esta variable se utilizó para llevar una cuenta de los puntos del jugador en cada partida.
* int ganapuntos = 0; Esta variable se utilizó para determinar en cuantos tiros el jugador ganó puntos.
* int pares = 0; Esta variable fue utilizada para determinar cuantas veces el dado del jugador caía en un número par.
* int impares = 0; Esta variable fue utilizada para determinar cuantas veces el dado del jugador caía en un número impar.
* int pares2 = 0; Esta variable fue utilizada para determinar cuantas veces el dado de la casa caía en un número par.
* int impares2 = 0; Esta variable fue utilizada para determinar cuántas veces el dado de la casa caía en un número impar.
* int iguales = 0; En esta variable se determinó cuantas veces los dados caían en un numero igual en el mismo tiro.
* int casagana = 0; Esta variable se utilizó para saber cuantas partidas ganaba la casa.
* int jugagana = 0; Esta variable se utilizó para saber cuantas partidas ganaba la casa.
* double probabilidad; Esta variable se utilizó para hacer una división y elevarla a la potencia del numero de tiros, para el porcentaje de probabilidad que tenía el jugador de ganar las partidas (siempre era el mismo).
* int dado1; Para guardar en una variable el valor del número random 1.
* int dado2; Para guardar en una variable el valor del número random 2.

Adicionalmente, se trabajaron las variables que el usuario tenía que ingresar dentro del programa:

* int partidass; Variable solicitada al usuario para determinar el número de partidas que se realizarían en el juego.
* int numtiros; Variable solicitada al usuario para determinar el número de tiros que tendría cada partida.

### SALIDAS

Dentro del programa se solicitaban las siguientes salidas como requisitos:

* El ganador de la partida: El jugador o la casa: se imprimió en la consola una comparación entre la variable int puntoscasa y la variable int puntosjuga para saber quién ganó la partida.
* Los tiros realizados en cada partida, cada dado por separado: dentro del ciclo for en donde se determinaba cada turno, se optó por imprimir el resultado de cada uno de los dados con las variables int dado1 e int dado2.
* Cuántos tiros ganó el jugador, es decir en cuantos tiros el ganó puntos: Se imprimió el resultado de la variable int ganapuntos.
* Cuál era su probabilidad de ganar: se imprimió la probabilidad calculada según las condiciones que tenía el juego previamente descritas. Se expresó con la variable double probabilidad además de que se trabajó en forma de porcentajes.
* En cuántos tiros le salieron números pares e impares: se imprimió en la consola la variable int pares.
* En cuántos tiros le salieron números iguales: se imprimió en la consola la variable int impares.
* Punteo final: se imprimió la variable int puntosjuga.

### PROBABILIDAD

Para trabaja la probabilidad de que el jugador ganara puntos se tomó solamente en cuenta las condiciones en donde el jugador podía ganar puntos. Por lo que se determine que quedaría así:

# DISEÑO:

### DIAGRAMA DE FLUJO

Diagrama, Esquemático

Descripción generada automáticamente

### DIAGRAMA DE CLASES

N/A, no aplica.

# CONCLUSIONES

1. Es necesario comprender las restricciones y reglas del juego para poder crear de una manera adecuada un algoritmo que refleje la realidad
2. Para realizar un programa de una acción repetitiva es importante repasar el orden de las acciones y determinar cuales de penden de las otras.
3. La documentación del proyecto fue importante para tomar en cuenta todos los lineamientos requeridos.

# RECOMENDACIONES

1. Es importante tomar en cuenta que la mejor forma para trabajar los vectores es con el ciclo for, es una gran herramienta.
2. Para trabajar juegos y restricciones en donde se implica una acción repetitiva ya se por funcionalidad del juego o por alguna funcionalidad requerida que aún no se cumple, los ciclos como el for y el while tienen mucha aplicación.
3. No es necesario usar tantas condiciones para determinar el orden si se programa justo en el orden como debería de ejecutarse el programa.

# REFERENCIAS

1. Universidad Rafael Landívar, Facultad de Ingeniería. Proyecto práctico No.2 Dados, 2023.