



Universidad Tecnológica

Centroamericana

FACULTAD DE INGENIERÍA EN SISTEMAS

MINIPROYECTO 2

JUAN MANUEL BORJAS FIGUEROA 12151124
EMILIO ALEJANDRO CANTARERO HERNANDEZ
12111141

LENGUAJES DE PROGRAMACION

MINIPROYECTO 2 ADIVINA LA COMBINACIÓN

TEGUCIGALPA M.D.C

18 de junio del 2023

1 Introducción

Como parte de la asignación Miniproyecto 2 de la clase de lenguajes de programación, se estableció el objetivo de la creación de un pequeño juego de adivinar números aleatorios, similar al popular Wordle. Para alcanzar este objetivo, se estableció que el lenguaje de programación a utilizar sería el lenguaje funcional F#. En el presente reporte se dan más detalles sobre su desarrollo, así como su arquitectura y algunos ejemplos de su funcionamiento.

2 Descripción del juego

Como se mencionó anteriormente, el juego consiste en adivinar números generados de forma aleatoria. El juego cuenta con tres dificultades diferentes, en las cuales la cantidad de intentos que el jugador tiene para acertar el número generado aumenta cada vez más con respecto a la dificultad anterior, así como el número de posibles incógnitas generadas y la variedad de dígitos que puedan contener también suben. Luego de elegir la dificultad, el usuario ingresa un número a forma de posible respuesta. El programa genera una retroalimentación estructurada así: Si un dígito de la respuesta generada, está en la misma posición que el número generado originalmente, se devolverá un '+'. Si un dígito de la respuesta ingresada está presente en el número original, pero no está en su posición correcta, se devolverá un '-'. Y si cualquier dígito de la respuesta ingresada, definitivamente no se encuentra en el número original, se devuelve una 'x'. El objetivo del usuario será guiarse de dicha retroalimentación para averiguar el número generado antes de quedarse sin intentos. Cabe mencionar que dicho número generado representa la combinación de una caja fuerte. Si el usuario se queda sin intentos, habrá perdido la partida, y la caja fuerte será destruida. Caso contrario, habrá ganado y la caja fuerte se abrirá.

3 Arquitectura de su Programa

1. "adivina" : Es la función principal del juego de adivinar números, se encarga de ejecutar el juego. Toma como parámetros la cadena que el jugador debe adivinar y el número máximo de intentos permitidos. En cada iteración del juego, solicita al jugador que ingrese un intento, compara el intento con la cadena a adivinar y proporciona retroalimentación al jugador. También controla el número de intentos y muestra mensajes de victoria o derrota.
2. "generarCadenaAleatoria" : Esta función genera una cadena aleatoria de números. Toma como parámetros la cantidad de dígitos de la cadena y el rango máximo de los dígitos. Utiliza la clase Random para generar los dígitos aleatorios y los concatena en una cadena.
3. "menu" : Esta función muestra un menú al jugador para que elija la dificultad del juego. Dependiendo de la opción seleccionada, genera una

cadena aleatoria y llama a la función adivina con la cadena y un número específico de intentos.

4. "Main" : Esta es la función principal del programa. Llama a la función menu para iniciar el juego y espera a que el jugador presione cualquier tecla para salir del programa.

4 Ejemplos

1. Dificultad Fácil:

```
Bienvenido al juego de adivinar números
Elige la dificultad:
1. Fácil
2. Intermedia
3. Difícil
1
Adivina el número:
1234
---x
Adivina el número:
0123
+--x
Adivina el número:
3624
+-xx
Adivina el número:
5327
++-x
Adivina el número:
7321
¡Felicidades! Adivinaste y has ganado.
++++
```

2. Dificultad Intermedia:

```
Bienvenido al juego de adivinar números
Elige la dificultad:
1. Fácil
2. Intermedia
3. Difícil
2
Adivina el número:
01234
-xxxx
Adivina el número:
25678
---xx
Adivina el número:
52768
+--xx
Adivina el número:
62575
++--x
Adivina el número:
52575
++--x
Adivina el número:
72257
++++x
Adivina el número:
72757
++++x
Adivina el número:
72755
+++--
Adivina el número:
72557
¡Felicidades! Adivinaste y has ganado.
+++++
|
```

3. Dificultad Difícil:

```
Bienvenido al juego de adivinar números
Elige la dificultad:
1. Fácil
2. Intermedia
3. Difícil
3
Adivina el número:
123456
+--xxx
Adivina el número:
024686
++--xx
Adivina el número:
204866
++--xx
Adivina el número:
406286
++++xx
Adivina el número:
376276
++-xxx
Adivina el número:
806286
++++xx
Adivina el número:
506886
+++++x
Adivina el número:
506386
¡Felicitades! Adivinaste y has ganado.
+++++
|
```

5 Análisis

5.1 ¿Como compararía implementar un juego de este tipo usando F# vs hacer una versión en un lenguaje orientado a objetos?

Diria que en un lenguaje orientado objetos la implementacion del juego seria algo mas natural y apegado al mundo real, se podria implementar una clase que fuera la que realizara las cadenas aleatorias de teniendo la organizacion del programa de manera mas eficaz, aunque sinceramente es la unica ventaja que

miro, ya que me parece que F# es un lenguaje bastante potente y sin mencionar que posee cosas de un lenguaje imperativo tales como los ciclos que hacen que viviendo de un lenguaje por así decirlo mas tradicional, todo sea mas intuitivo

5.2 ¿Hubo partes del juego que son más fáciles de implementar en F# que usando lenguajes como Java?

La cosa que mas nos llamo la atencion fue el uso de las funciones lambda, o sobre la marcha, las cosidero bastante faciles de aprender y sobre todo de implementar por lo tanto hizo que la realizacion de la logica de la adivinanza fuera mucho mas facil, sin mencionar que este lenguaje posee cosas integradas para iterar sobre las cadenas de manera rapida y eficiente.

5.3 Si tuviese más tiempo, ¿qué otras características le adicionarían al juego?

Le adicionaríamos mas niveles y un nuevo sistema de pistas que permitiera que el usuario recibiera feedback de las posiciones incorrectas pero mediante el uso de acertijos.

5.4 Cualquier otro comentario u observación relevante.

Nos llamo la atencion que al ambos estar familiarizados con C# y Rust se nos hizo super facil aprender F# ya que es como si hiciera una combinacion de ambos lenguajes pero facilita aun mas las cosas siendo de alto nivel, sin mencionar que es un lenguaje de tipado dinamico que siempre es algo que se agradece ya que se ahorra mas tiempo al momento de declarar las variables.

6 Dificultades Encontradas

1. Se nos hizo algo extraña la asignacion de un nuevo valor para una variable mutable
2. Tratamos de iterar dos veces, es decir una iteracion dentro de otra en las cadenas pero era algo que nos pareció casi imposible por lo que recurrimos al uso de ciclos for.

7 Conclusiones

F# es un lenguaje bastante agradable al momento de programar, es muy conveniente para personas que talvez no tengan tanta experiencia programando ya que en nuestro caso sin haber visto o usado algo del mismo, se nos hizo muy intuitivo ya que incorpora cosas de otros lenguajes populares.