



Anthony Barrantes Jiménez – 2023152240

Instituto Tecnológico de Costa Rica – Campus Central Cartago

Escuela de Computación

Bachillerato en Ingeniería en Computación

Taller de Programación – Grupo 04

Profesor William Mata Rodríguez

Programa 2 – Juego De Lógica Kakuro

I Semestre 2023

28 de mayo de 2023

Contenido

Enunciado del proyecto	3
Temas investigados.....	3
<i>Programación dirigida por eventos.....</i>	<i>3</i>
<i>Uso de interfaz gráfica de usuario (GUI: Graphical User Interface) en el desarrollo de software</i>	<i>3</i>
<i>Desarrollo de GUI con Tkinter.....</i>	<i>4</i>
Conclusiones del trabajo	5
Estadística de tiempos	5
Referencias.....	6
Lista de revisión del proyecto	7

Enunciado del proyecto

El proyecto consiste en el desarrollo de un juego matemático de interfaz gráfica basado en la lógica. El nombre de este juego es Kakuro y su objetivo es el de rellenar filas y columnas que sumen un valor específico indicado. El jugador puede elegir un número del 1 al 9 y colocarlo en cualquiera de las casillas vacías del tablero 9x9. Los números bajo la línea por medio de la casilla indican la suma de la columna y los de arriba deben ser la suma de la fila.

Kakuro es un juego similar al conocido Sudoku, mas difiere en algunas de sus reglas, una de ellas es la posibilidad de tener el mismo número en una fila o una columna siempre y cuando no formen parte de la misma suma vertical u horizontal. El juego será dividido en 4 dificultades (Fácil, Medio, Difícil y Experto) y 3 modos (Cronometro, Sin reloj y Timer), todas ellas configurables en su respectivo menú.

Temas investigados

Programación dirigida por eventos

La programación dirigida a eventos es una estrategia de programación que no se basa en la ejecución secuencial de instrucciones, como puede ser la programación estructurada convencional, mas se basa en la escucha de señales que, al cambiar su estado, pueden desencadenar una serie de acciones en el programa.

Este paradigma de programación es altamente útil en el campo de interfaces gráficas ya que permite desencadenar funciones al presionar un botón, ingresar un dato, tocar un menú o al llevar a cabo cualquier acción.

Podemos entender un programa dirigido a eventos como un ciclo, cuyas funciones son ejecutadas cuando se cumple una cierta condición específica. Así imaginamos cada ventana mostrada en pantalla como un ciclo en constante ejecución cuyos resultados varían dependiendo de lo que nosotros, los usuarios, hagamos con la misma.

La programación dirigida a eventos la encontramos en una variedad de aplicaciones, desde interfaces gráficas (donde tenemos elementos como botones esperando a cambiar su estado para ejecutar una acción, hasta robótica (donde tenemos sensores y otros elementos esperando a ser desencadenados para ejecutar su función o funciones) o sistemas de alarmas (donde tenemos un sensor o una serie de sensores que activaran una alarma si son manipulados).

Uso de interfaz gráfica de usuario (GUI: Graphical User Interface) en el desarrollo de software

Una interfaz gráfica de usuario o GUI (por sus siglas en inglés: Graphical User Interface) es una herramienta usada por una aplicación para comunicarse en el usuario.

Las GUI hacen uso de una serie de imágenes y objetos gráficos que representan tanto la información destinada al usuario como las acciones que el mismo puede elegir realizar.

Las GUI no son estructuras cerradas, todo lo contrario, son altamente dinámicas y su contenido o elementos varían dependiendo de la aplicación necesaria. Estas interfaces pueden ser creadas con una variedad de lenguajes de programación, en algunos de manera nativo y en otros con el uso de módulos (como Python con su módulo Tkinter).

Las interfaces gráficas tienen un amplio rango de usos, prácticamente todas las aplicaciones que visualizamos en una máquina hacen uso de una GUI, desde los navegadores web hasta el editor de texto usado para la elaboración de esta documentación o el IDE usado para escribir el código del programa. Las GUI son un elemento primordial en el uso moderno de una computadora, son las que nos dan una conexión directa entre lo que ejecutamos y lo que vemos.

Desarrollo de GUI con Tkinter

Tkinter es un módulo integrado en Python para la creación de interfaces gráficas de usuario, aunque no es el componente de creación de ventanas más accesible, es por lo general la mejor opción para el desarrollo de GUIs en este lenguaje de programación. Esta librería cuenta con una variedad inmedible de elementos que nos facilitan las tareas requeridas, los objetos usados en el presente proyecto se detallan a continuación:

- Clase Tk: esta clase nos permite crear un objeto que será la ventana donde ejecutaremos nuestro programa principal. Como se indico anteriormente, las ventanas son ciclos y por ende debemos usar su método “mainloop” para iniciar su ejecución. Algunos otros métodos usados sobre este objeto son title (nos permite definir el título de la ventana), geometry (nos permite establecer las dimensiones de la ventana), {min/max}size (nos permite establecer los tamaños mínimos y máximos de la ventana) y iconbitmap (nos permite establecer el ícono de la ventana).
- ImageTk.PhotoImage: nos permite leer un archivo de imagen de tal modo que pueda ser usado dentro de la ventana
- Label: nos permite desplegar texto o imágenes, es una etiqueta cuyo contenido podemos modificar dependiendo lo que necesitemos.
- Método place: es un método de todos los elementos de Tkinter, nos permite desplegar los elementos que creemos en una posición (x, y) específica.
- Button: nos permite crear un botón personalizable, sus atributos varían desde estéticos (text, bg, height o width) hasta funcionales (command, state). La función incluida como argumento de command será ejecutada cuando se presione el botón.
- MessageBox: usado para desplegar pequeñas ventanas con mensajes informativos, de error o interrogativos.
- Toplevel: clase usada para crear ventanas que están relacionadas al mainloop o ventana principal, mas son desplegadas como una ventana separada. Comparte la gran mayoría de métodos con la clase Tk

- Entry: nos permite crear una caja de entrada de texto, podremos obtener su texto y usarlo como comunicación entre el usuario y el programa. Este objeto nos permite la recolección de datos personalizados por el usuario.
- OptionMenu: nos permite crear menús “dropdown” en los cuales el usuario podrá elegir entre una serie de opciones que estén dentro de un contenedor. Este objeto tiene una alta utilidad en la elección de una serie de configuraciones u otras opciones.
- StringVar: variable de tipo string usada especialmente para contener la opción actual de menús como el OptionMenu.

Conclusiones del trabajo

Afrontar la creación de una GUI desde cero puede ser un desafío, mas una buena esquematización y un planeamiento adecuado previo al inicio del desarrollo facilita este proceso de manera considerable. Una investigación sobre los objetos, clases y estructuras usadas por Tkinter es esencial para poder escribir un código funcional y eficiente.

Es importante mantener una mente abierta durante el proceso de desarrollo, ya que muchas veces tendremos que hacer uso de estructuras u organizaciones previamente desconocidas o poco intuitivas; si no se está abierto al cambio, el código no será óptimo y, en algunos casos, inservible.

Es común que el desarrollador tenga cierto bloqueo creativo al crear la GUI, si esto pasa, es importante usar el material de referencia como guía de diseño para mantener una estética deseada con las funcionalidades requeridas. Una muy buena práctica es obtener los colores a usar en el programa del documento de especificaciones del proyecto.

Como cualquier proyecto, el aprendizaje es amplio y ciertamente útil, pero en este proyecto podemos destacar principalmente los conocimientos adquiridos en materia de diseño de GUIs, construcción de GUIs en Tkinter, uso de programación dirigida por eventos y creación de recursos gráficos (logos, íconos, entre otros)

Estadística de tiempos

Actividad realizada	Horas destinadas
Análisis del problema planteado en el proyecto	1
Diseño de algoritmos	2
Investigación previa	1
Investigación durante el desarrollo	2
Creación de recursos gráficos	0.5
Programación	15
Documentación interna	1.5
Pruebas	1

Elaboración del manual de usuario	1
Elaboración de la documentación	2
Total	27

Referencias

Colaborador De TechTarget. (2017, marzo). *Interfaz gráfica de usuario o GUI*.

ComputerWeekly.es. Recuperado 27 de mayo de 2023, de

<https://www.computerweekly.com/es/definicion/Interfaz-grafica-de-usuario-o-GUI>

Villafranz, C., & Barrios, V. (s. f.). *Programación dirigida por eventos* (De H. Vega).

Platzi. Recuperado 27 de mayo de 2023, de [https://platzi.com/clases/2211-](https://platzi.com/clases/2211-historia-programacion/35079-programacion-dirigida-por-eventos/)

[historia-programacion/35079-programacion-dirigida-por-eventos/](https://platzi.com/clases/2211-historia-programacion/35079-programacion-dirigida-por-eventos/)

Lista de revisión del proyecto

Concepto	Puntos	Avance 100%/0	Puntos obtenidos	Análisis de resultados
Opción Jugar: despliegue ventana de juego	20	100%		
Botón Iniciar Juego (incluye creación del archivo del Top 10)	15	100%		
Botón Deshacer Jugada	8	100%		
Botón Rehacer Jugada	8	100%		
Botón Borrar Casilla	2	100%		
Botón Borrar Juego	2	100%		
Botón Terminar Juego	5	100%		
Botón Top 10	5	100%		
Botón Guardar Juego	10	100%		
Botón Cargar Juego (incluye el despliegue del mismo)	10	100%		
Opción Configurar	5	100%		
Ayuda en el programa: Manual de usuario	5	100%		
Cronómetro o Timer tiempo real	5	100%		
TOTAL	100	100%		