|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **INSTRUMENTOS DE EVALUACIÓN** | | |  |
| Código: FO-FIT-129 | Versión: 001 | Página 1 de 1 |

|  |  |
| --- | --- |
| **Instrumento de Evaluación:** | Evaluación final - Diseño de Algoritmos III |
| **Nombre de Estudiante:** | José Daniel Rodríguez López |
| **Guía de Aprendizaje:** | Diseño de Algoritmos III |
| **Fecha:** | 03 de Octubre de 2023 |
| **Jornada:** | Noche |
| **Nombre del Docente:** | James Mosquera |

De acuerdo con el último digito de su documento de identidad le quedara asignado el ejercicio correspondiente

|  |  |
| --- | --- |
| N° | Ejercicio |
| 0 | **Buscaminas:** Crea una versión simple del juego de buscaminas donde los jugadores intentan descubrir las minas ocultas en un tablero sin hacer estallar ninguna. |
| 1 | **Cronómetro o Temporizador**: Diseña un cronómetro o temporizador que permita a los usuarios medir el tiempo. |
| 2 | **Conversor de Unidades**: Crea una aplicación que convierta entre diferentes unidades, como conversor de monedas, de unidades de medida, etc. |
| 3 | **Generador de Contraseñas**: Diseña un generador de contraseñas que cree contraseñas seguras de forma aleatoria. |
| 4 | **Calendario**: Implementa un calendario simple que muestre los eventos programados para un día específico. |
| 5 | **Lista de Tareas**: Crea una aplicación de lista de tareas donde los usuarios puedan agregar, editar y eliminar tareas pendientes. |
| 6 | **Gestor de Gastos**: Desarrolla una aplicación que permita a los usuarios registrar sus gastos y calcular su presupuesto. |
| 7 | **Juego de Adivinanza**: Crea un juego donde el programa selecciona un número aleatorio y el usuario tiene que adivinarlo. |
| 8 | **Calculadora de IMC**: Implementa una calculadora de índice de masa corporal (IMC) que ayude a los usuarios a evaluar su salud. |
| 9 | **Gestor de Contactos**: Diseña una aplicación que permita a los usuarios gestionar su lista de contactos. |

Nota Aplica los conocimientos vistos en clase

<https://github.com/drl17/DisenoAlgIIIProy.git>

#Esta es la versión final del aplicativo

#Importamos las libreriar para el diseño de aplicativos en web, para invocar datos aleatorios y las ventanas con mensajes

import tkinter

import random

from tkinter import messagebox

#Creamos la interfaz con sus respectivas dimensiones

Interfaz=tkinter.Tk()

Interfaz.geometry("480x374")

#Agregamos una imagen de fondo para la interfaz

Fondo\_Interfaz=tkinter.PhotoImage(file="acertijo.png")

Fondo\_Imagen=tkinter.Label(Interfaz,image=Fondo\_Interfaz)

Fondo\_Imagen.pack()

#Obtenemos la primera variable como el dato random que elige la máquina

ganador = random.randint(1, 10)

#Agregamos un texto de bienvenida

Texto\_Bienvenida=tkinter.Label(Interfaz, text="¡Hola, bienvenidos al juego del acertijo!")

Texto\_Bienvenida.place(x=130, y=100)

#Agregamos un texto explicando el juego

Instruccion1=tkinter.Label(Interfaz, text="Ingresa un número [1 al 10] y comprueba si adivinas el número secreto. Tienes 3 intentos")

Instruccion1.place(x=4, y=150)

#Creamos un cuadro de texto para que la interacción del usuario y donde ingresará el número que buscará adivinar

Caja\_Texto=tkinter.Entry(Interfaz)

Caja\_Texto.place(x=170, y=200)

#Definimos la función principal donde comparamos el número obtenido por la máquina y el número ingresado por el usuario

#Según los escenarios definimos unos resultados

Contador\_Intentos=0

def comprobar():

    global Contador\_Intentos

    Contador\_Intentos += 1

    if Contador\_Intentos <= 3:

        a = int(Caja\_Texto.get())

        if a == ganador:

            messagebox.showinfo("¡Ganaste!", "Has derrotado al acertijo")

            Interfaz.destroy()

        elif a > ganador:

            messagebox.showwarning('Fallaste', f'Te quedan {3 - Contador\_Intentos} intentos. Tu intento ha sido alto.')

        else:

            messagebox.showwarning('Fallaste', f'Te quedan {3 - Contador\_Intentos} intentos. Tu intento ha sido bajo.')

    else:

        messagebox.showerror("Game Over", "Has agotado todos tus intentos.")

        Interfaz.destroy()

#Creamos un botón para ejecutar la función principal y la asociamos

Boton\_Comprobar=tkinter.Button(Interfaz, text="Comprobar", command=comprobar)

Boton\_Comprobar.place(x=200, y=250)

#Creamos un botón para cerrar cuando haya finalizado el juego

Boton\_Cerrar=tkinter.Button(Interfaz, text="Cerrar", command=Interfaz.destroy)

Boton\_Cerrar.place(x=215, y=300)

Interfaz.mainloop()

|  |  |  |  |
| --- | --- | --- | --- |
| **N.º** | **Aspecto para evaluar** | **Si** | **No** |
| **1** | Leer y comprender bien los problemas algorítmicos |  |  |
| **2** | Define bien las constantes y variables |  |  |
| **3** | Emplea bien la nemotécnica de variables dentro de las funciones |  |  |
| **4** | Identifica y aplica las distintas herramientas de programación vistas |  |  |

Firma del estudiante Firma del instructor