| **FACULTAD/DEPENDENCIA:**  Ingeniería | | | **FECHA DE ENTREGA** | | | |
| --- | --- | --- | --- | --- | --- | --- |
| **PROGRAMA ACADÉMICO/ ÁREA:**  TECNOLOGÍA EN SISTEMAS | | | **DD**  27 | | **MM**  08 | **AAAA**  2023 |
| **ASIGNATURA**:  Construcción de software IV | | **PROFESOR:**  Juan Guillermo Duque Galvis | | | | |
| **TIPO DE EVALUACIÓN** | Práctica | **PORCENTAJE:**  15% | | **CALIFICACIÓN:** | | |
| **ALUMNOS:** | | | | | | |

**Entrega**: Realizar los ejercicios propuestos y subirlos a la plataforma [GitHub.com](https://github.com).   
Solo el archivo Program.cs

Enviar este documento con los integrantes (nombres y apellidos con documento) Y enlace del repositorio al correo [jduqueg2@tdea.edu.co](mailto:jduqueg2@tdea.edu.co)

**Lenguaje:** C#

**Ayudas**:

[Visual Studio Code con GitHub. Subiendo y sincronizando proyecto](https://www.youtube.com/watch?v=ngow7sPfSDQ)

**Ejercicios**:

**Ejercicio 1: Calculadora de Fracciones**

**Objetivo**: Crear una calculadora que maneje fracciones. El usuario ingresará dos fracciones y un operador. La calculadora deberá realizar la operación. Debes manejar sumas, restas, multiplicaciones y divisiones. Si el usuario intenta dividir entre cero, deberás mostrar un mensaje de error.

**Ejercicio 2: Generador de Tablas de Multiplicar**

**Objetivo**: El usuario proporcionará un rango (por ejemplo, del 3 al 7). El programa generará tablas de multiplicar para cada número en ese rango. Por ejemplo, si el rango es del 3 al 5, el programa generará la tabla del 3, la tabla del 4 y la tabla del 5. Sin embargo, existe un giro: para cada tabla, el programa ocultará aleatoriamente uno de los resultados y desafió al usuario a completarlo. Por ejemplo, "3 x ? = 21" y el usuario debe adivinar que el número faltante es 7.

**Ejercicio 3: Número especial**

**Objetivo**: Crear un programa que determine si un número ingresado por el usuario es un número "especial". Un número es "especial" si cumple con los siguientes criterios:

1. Es divisible entre 5.
2. No es divisible entre 2 ni 3.
3. La suma de sus dígitos es mayor a 10.

**Ejercicio: Adivinanza de Frase Oculta**

**Objetivo**: Desarrollar un programa que presente al usuario una frase con palabras ocultas y permita al usuario adivinar las palabras faltantes. El programa debe proporcionar retroalimentación sobre si la palabra ingresada corresponde a alguna de las palabras ocultas.

Instrucciones:

La frase original es: "El gato juega en el jardín".

La frase que se muestra al usuario con palabras ocultas es: "El \_\_\_\_\_ juega \_\_\_\_\_ el \_\_\_\_\_".

* El usuario tiene 10 intentos en total para adivinar las 3 palabras ocultas.
* Por cada intento, el usuario ingresa una palabra.
* Si la palabra ingresada corresponde a alguna de las palabras ocultas y esa palabra no ha sido adivinada anteriormente, informar al usuario que ha acertado.
* Si la palabra ingresada es incorrecta o ya fue adivinada previamente, informar al usuario que está incorrecto y restar un intento.
* Si el usuario adivina las 3 palabras antes de agotar sus intentos, mostrar un mensaje de felicitación y revelará la frase completa.
* Si el usuario agota todos sus intentos sin adivinar todas las palabras, informarle que ha perdido y finalizar el juego.
* Consejo: Para simplificar el desarrollo, no es necesario considerar variaciones en mayúsculas/minúsculas. Es decir, puedes asumir que "gato" y "Gato" serían respuestas diferentes.