SEP
SECRETARÍA DE EDUCACIÓN PÚBLICA

TECNOLÓGICO NACIONAL DE MÉXICO
INSTITUTO TECNOLÓGICO SUPERIOR DE MONCLOVA
"EJÉRCITO MEXICANO"
GOBIERNO DEL ESTADO DE COAHUILA

CAMPUS MONCLOVA

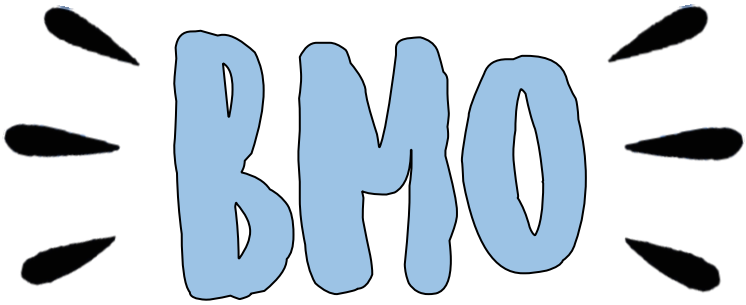
PROYECTO FINAL PROGRAMACION ORIENTADA A OBJETOS

Daniela Edith García Muñoz

Informática 2do semestre

Índice

- BMO
- Componentes de Windows form
- The Arcade
- Componentes de consola



BUENOS MENSAJES OCULTOS EN WINDOWS FORM

El programa es un simulador para cifrar y descifrar mensajes usando los métodos Cesar y atbash,

La cristología es el ámbito que se ocupa de las técnicas de cifrado o codificado, la cristología significa literalmente escritura secreta y este es el objetivo principal de este programa.

El método cesar también conocido como cifrado por desplazamiento, es una de las técnicas de cifrado más simples y más usadas. (Wikipedia, 2005) Fue creado por Julio Cesar y este se relaciona con el cifrado de vi generé, la transformación en el método cesar puede ser variada pero la que yo uso en este simulador es de 4 espacios hacia la derecha quedando de la siguiente manera:

Texto original: ABCDEFGHIJKLMNOPQRSTUVWXYZ

Texto codificado: DEFGHIJKLMNOPQRSTUVWXYZABC

El método atbash también se encuentra en este simulador Pertenece a la llamada criptografía clásica y es un tipo de cifrado por sustitución. (Wikipedia, 2005) Se le denomina también método de espejo, pues consiste en sustituir la primera letra (A) por la última (Z), la segunda (B) por la penúltima (Y) quedando de la siguiente manera:

Texto original: ABCDEFGHIJKLMNOPQRSTUVWXYZ

Texto codificado: ZYXWVUTSRQPONMLKJIHGFEDCBA

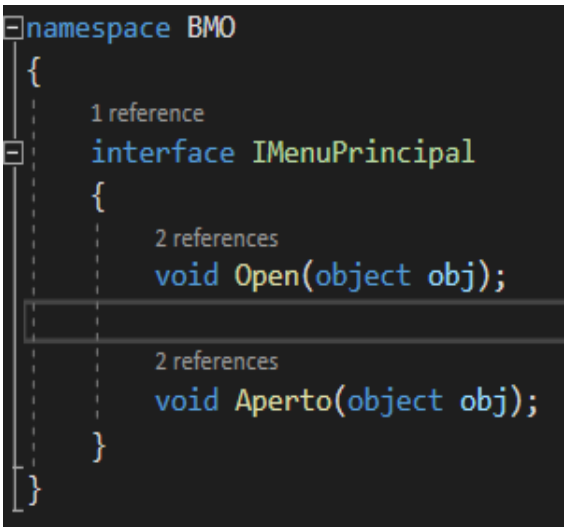
CONTIENE

El código empieza desde el frmPrincipal en este formulario el cual contiene una progressbar la cual se empieza a llenar cuando presionas el botón comenzar.

Cuando la barra este llena esta abrirá el siguiente formulario el cual es frmMenu, aquí tienes la opción de elegir con cual método quieres hacer tu cifrado o descifrar algunas frases que ya se encuentran en estos criptogramas, el form contiene 3 botones

- Atbash: el cual hace abre frmAtbash
- Cesar: Abre frmCesar
- Salir: este simplemente hace lo que dice, se sale del programa

En este programa existen dos interfaces una de ellas es IMenuPrincipal la cual cuenta con dos métodos:



Cuenta con una clase llamada OpcionesMenu la cual hereda de frmMenu y IMenuPrincipal se encuentra implementándola esta clase contiene los métodos que hacen que frmAtbash y frmCesar puedan abrir

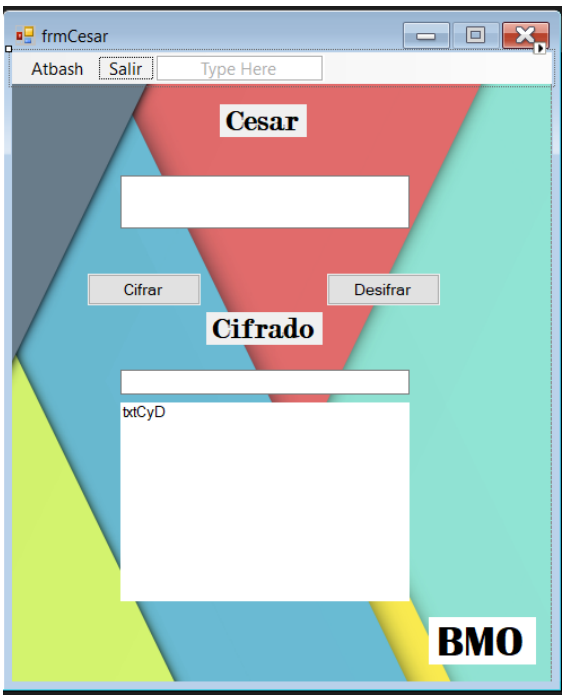
OpcionesMenu y frmMenu se encuentran en una carpeta llamada Mu

Otra carpeta es Cesar en ella se encuentra frmCesar y dos clases llamadas MetodoCesar y MetodomnsCesar, las dos heredan de frmCesar pero esta última es implementada por la interfaz IMenu la cual contiene el método abrir

MetodomnsCesar lo que hace es darle memoria al método de la interfaz IMenu

En la clase Metodo cesar contiene parámetros y propiedades las cuales logran hacer que las letras de lo que escribes se desplacen 3 lugares de esta manera creando el cifrado, también cuenta con un método que hace que se recorran 3 lugares más adelante si lo que quieres es descifrar el código.

Logrando así que estas dos clases se junten en frmCesar el cual contiene 2 textbox (en uno escribes y en el otro descifra o cifra lo que escribas) 1 listbox (se encarga de guardar todo lo que este escribiendo en los textbox) dos botones (uno para cifrar y otro para descifrar) y por ultimo un menustrip (que cuenta con la opción de irte a frmAtbash o salir del programa)



Ya por ultimo nos encontramos con la carpeta de Atbash la cual contiene frmAtbash y las dos clases MetodoAtbash y MetodomsnAtbash, en este caso solo la segunda hereda de frmAtbash y esta implementada por la interfaz IMenu que al igual que con MetodomsnCesar hace que abra un form solo que aquí en lugar de abrir frmAtbash abre frmCesar.

La clase MetodoAtbash transforma el método string en un arreglo de char para crear el encapsulamiento, de esta manera el cifrado podrá llevarse a cabo.

Cuenta con casi los mismos elementos que frmCesar pero aquí solo se encuentra un botón ya que este es un método de espejo no importa si estas cifrando o descifrando se usan las mismas variables

THE ARCADE

La función de The Arcade es muy sencilla, en él se encuentran dos juegos

El ahorcado el cual consiste en que tienes que adivinar la palabra o frases que este antes de que el muñeco quede totalmente dibujado

¿Qué numero? Aquí tienes que adivinar un número del 1 al 50 el cual es elegido al azar.

CONTIENE

Este programa contiene 2 carpetas, 4 clases y 2 interfaces

Todo empieza en programs.cs la cual contiene el Main y aquí este el método que hace que inicie la siguiente clase la cual es MenuPrincipal, el cual fue enseñado anteriormente

MenuPrincipal cuenta con un switch el cual contiene 3 casos

- caso 1 inicia el juego del Ahorcado
- caso 2 inicial el juego de Que numero?
- Y el caso 3 sale del programa

Nuestra primera carpeta se llama El ahorcado contiene 1 interfaz llamada IAhorcado y una clase llamada Juegol

En IAhorcado se encuentran los siguientes métodos:

```
namespace The_Arcade.El_ahorcado
{
    7 references
    interface IAhorcado
    {
        2 references
        string ElegirPalabra();

        2 references
        char[] GenerarEspacios(string palabra);

        2 references
        void ImprimirEspacios(char[] longitud);

        2 references
        bool CompruebaUna(string letra);

        2 references
        void ImprimirFinJuego(string palabra, int intentos, bool gano);
    }
}
```

Juegol Es implementada por esta interfaz y aparte de tener esos métodos con sus variables también tiene otros métodos llamados

- ImprimirHorca
- AbrirMenu
- Empezar

Y por último tenemos la carpeta Que numero la cual al igual que la carpeta pasada contiene 1 clase y 1 interfaz

IQueNumero nombre de la interfaz contiene estos métodos

```
namespace The_Arcade.Que_numero
{
    1 reference
    interface IQueNumero
    {
        2 references
        void Start();
        2 references
        void Abrir();
    }
}
```

IQueNumero implementa a la clase Juego2 la cual pone variables y parámetros para que se cree un numero al azar y de esta manera te dice que trates de adivinarlo

```
2 references
public void Start()
{
    Console.Clear();
    int Min = 1, Max = 51;
    Console.Title = "QUE NUMERO?";
    Console.WriteLine("Bienvenido a Que numero?! \n\n Voy a pensar un numero
      del " + Min + " al " + (Max - 1) + " y debes adivinarlo!");
    Console.Write("Escribe un numero: ");
    Random r = new Random(Guid.NewGuid().GetHashCode());
    int numeroElegido = r.Next(Min, Max);
    int numeroIngresado = 0;

    do
    {
        numeroIngresado = Convert.ToInt32(Console.ReadLine());

        if (numeroIngresado > numeroElegido)
        {
            Console.WriteLine("Es un numero menor a ese");
        }
        else if (numeroIngresado < numeroElegido)
        {
            Console.WriteLine(" Es un numero mayor a ese");
        }
        else
        {
            Console.WriteLine("Ganaste");
            Console.ReadKey();
        }
    } while (numeroElegido != numeroIngresado);
    Console.ReadLine();
    Abrir();
}
```

Al acabar Juego1 y Juego2 con solo apretar la tecla enter regresan al menú principal dándote oportunidad de repetir el juego, elegir otro o salir de la aplicación.