**Examen final de laboratorio**

**Instrucciones:** Cree un proyecto nuevo en C# con nombre *ExamenFinal\_Apellido\_Nombre.* En un único formulario, y creando clases donde se le indique, resolver tres de los cuatro problemas presentados a continuación. Suba su solución al espacio correspondiente en el portal antes de las 12:30 pm.

Antes de resolver un problema y aunque no se lo pidan, recuerde analizar sus entradas, procesos y salidas. Siempre es buena práctica también diseñar el diagrama de flujo de su solución. Aunque sean bosquejos, con ambas herramientas, la codificación deberá ser mucho más sencilla. ¡Éxitos!

1. Resolver la siguiente serie matemática calculando las potencias como multiplicaciones.

*(1 / (2^1)!) + (1 / (2^2)!) + (1 / (2^3)!) + ……+ (1 / (2^N)!)*

Utilizando los siguientes pasos:

* Cree una clase llamada OPERACIONES que contenga una función llamada Factorial y que calcule el factorial para un número N dado.
* Dentro de la clase cree otra función llamada Potencial que calcule la potencia de un número N dado haciendo uso de multiplicaciones. (Recuerde que 2^3 es igual a 2\*2\*2)
* Utilice ambas funciones dentro de un algoritmo para calcular el resultado de la serie a partir de un número entero positivo N dado.

**Análisis:**Entradas: Un número N ingresado por el usuario

Salidas: El resultado de la seria matemática anteriormente presentada

Procesos: Calcular el factorial del resultado dado de dividir 1 entre 2 elevado a una potencia

Restricciones: Tiene que ser un número entero (N)

1. Dos números amigos son dos números enteros positivos a y b tales que a sea la suma de los divisores propios de b, y b sea la suma de los divisores propios de a.
   1. Nota. La unidad (1) se considera divisor propio, pero no lo es el mismo número.

Un ejemplo es el par de números naturales (220, 284), ya que:

* Los divisores propios de 220 son 1, 2, 4, 5, 10, 11, 20, 22, 44, 55 y 110, que suman 284
* Los divisores propios de 284 son 1, 2, 4, 71 y 142, que suman 220.

Utilizando los siguientes pasos:

* Cree una clase llamada OPERACIONES, que contenga un método que calcule la sumatoria de los divisores de un número entero n dado.
* Dentro de la clase cree otro método que reciba dos números enteros y compare si son iguales o no.
* En el formulario principal debe crear un objeto de la clase OPERACIONES que se llame Amigos y utilice ambos métodos en un algoritmo que indique si 2 números son amigos o no. El resultado debe devolverse en un Messagebox.

**Análisis:**

Entradas: Dos números enteros

Salidas: Un mensaje indicando si el número previamente ingresado es un número amigo o no

Procesos: Determinar si la suma de los divisores del número da como resultado el otro número, y hacer lo mismo con el segundo número.

Restricciones: Deben ser números enteros.

1. En el formulario principal cree un vector de enteros de ocho posiciones, realizar un algoritmo que lea y almacene 8 números en él. Luego, en un DataGridView, desplegarlo en orden inverso (de atrás para adelante).

Ejemplo:

Leer y guardar en vector:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |

Desplegar en DataGridView:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |

**Análisis:**

Entradas: 8 números ingresados por el usuario  
Salidas: Desplegar los 8 números ingresado en un DataGridView pero en orden inverso (de atrás para adelante)  
Procedimientos: Asignar el valor a cada posición dentro del arreglo y luego ponerlos en la posición contraria.  
Restricciones: Los 8 números ingresados deben ser números enteros.