



UNIVERSIDAD ISRAEL

CIENCIAS DE LA INGENIERÍA

CARRERA DE SISTEMAS DE INFORMACIÓN

PLATAFORMAS DE DESARROLLO 1

SEMESTRE 2020 B

INFORME DE LABORATORIO 1-S1

TEMA: Conceptos Introdutorios de Visual Studio

ESTUDIANTE(S): Marco Ayala

CURSO: Séptimo

PARALELO: "A"

PROFESOR: Mg. Luis Fernando Aguas Bucheli

QUITO, 2020

1. TEMA: Conceptos Introdutorios de Visual Studio

2. OBJETIVOS:

- Conocer las características y alcances de Visual Studio
- Aprender los componentes y su entorno de trabajo
- Aplicar programación orientada a objetos
- Crear aplicaciones básicas hasta mas complejos

3. INTRODUCCION:

C# es un lenguaje fuertemente tipado. Todas las variables y constantes tienen un tipo, al igual que todas las expresiones que se evalúan como un valor. Cada declaración del método especifica un nombre, un número de parámetros, un tipo y una naturaleza (valor, referencia o salida) para cada parámetro de entrada y para el valor devuelto. La biblioteca de clases .NET define un conjunto de tipos numéricos integrados, así como tipos más complejos que representan una amplia variedad de construcciones lógicas, como el sistema de archivos, conexiones de red, colecciones y matrices de objetos, y fechas. Los programas de C# típicos usan tipos de la biblioteca de clases, así como tipos definidos por el usuario que modelan los conceptos que son específicos del dominio del problema del programa.

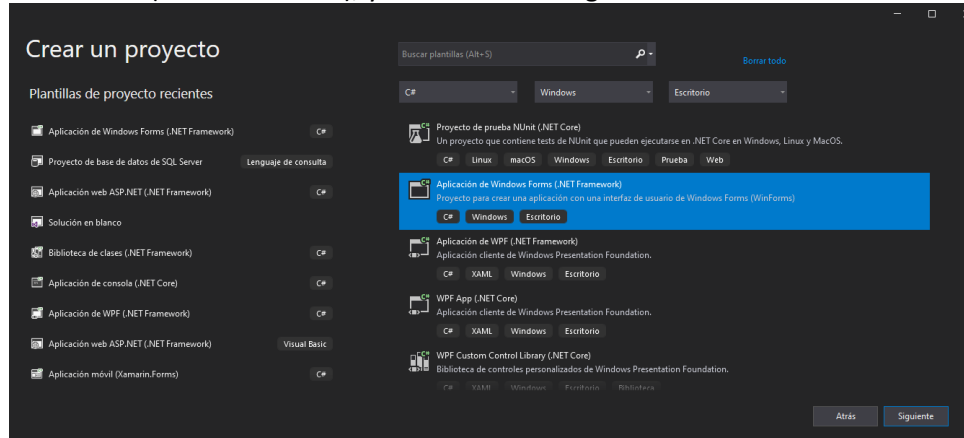
Entre la información almacenada en un tipo se pueden incluir los siguientes elementos:

- El espacio de almacenamiento que requiere una variable del tipo.
- Los valores máximo y mínimo que puede representar.
- Los miembros (métodos, campos, eventos, etc.) que contiene.
- El tipo base del que hereda.
- Interfaces que implementa.
- La ubicación donde se asignará la memoria para variables en tiempo de ejecución.

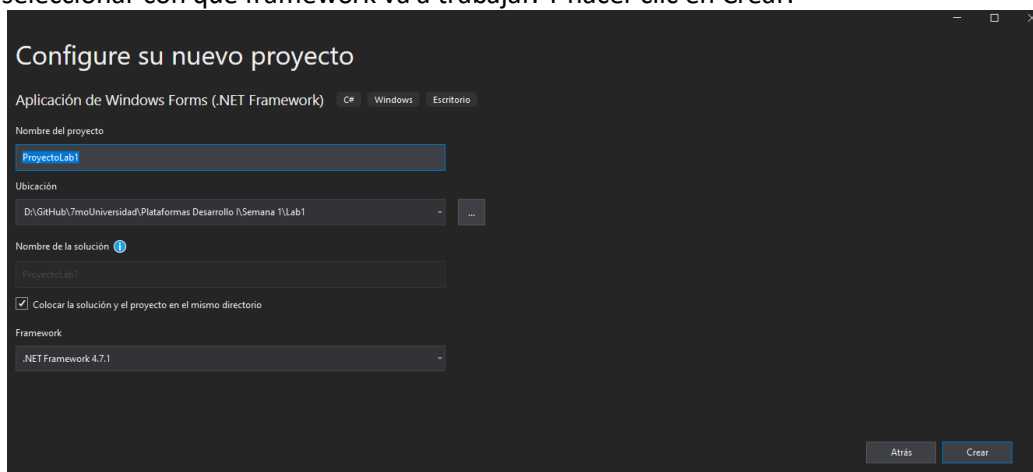
Los tipos de operaciones permitidas. El compilador usa información de tipo para garantizar que todas las operaciones que se realizan en el código cuentan con seguridad de tipos.

4. DESARROLLO:

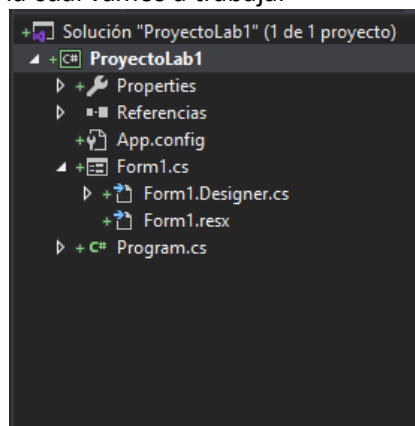
Al abrir Visual Studio en su versión 2019 nos ubicamos en la opciones de Aplicaciones de Windows Forms (Net Framweork), y seleccionamos siguiente.



En esta pantalla debemos ingresar el nombre del proyecto, ubicar la carpeta de la solución y seleccionar con que framework va a trabajar. Y hacer clic en Crear.



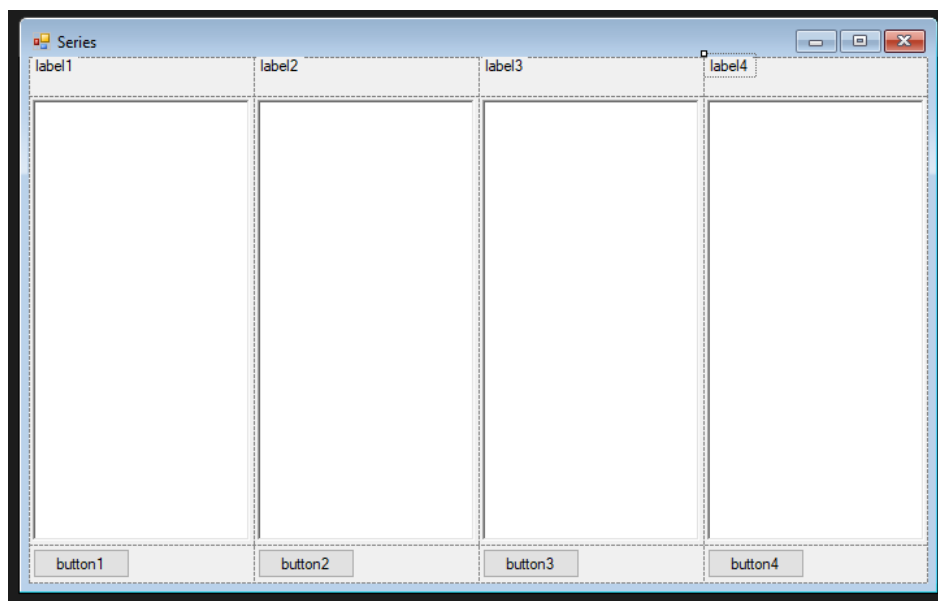
El contendrá estas carpetas y la siguiente estructura, una solución y una pantalla Inicial nombrada por Form1.cs en la cual vamos a trabajar



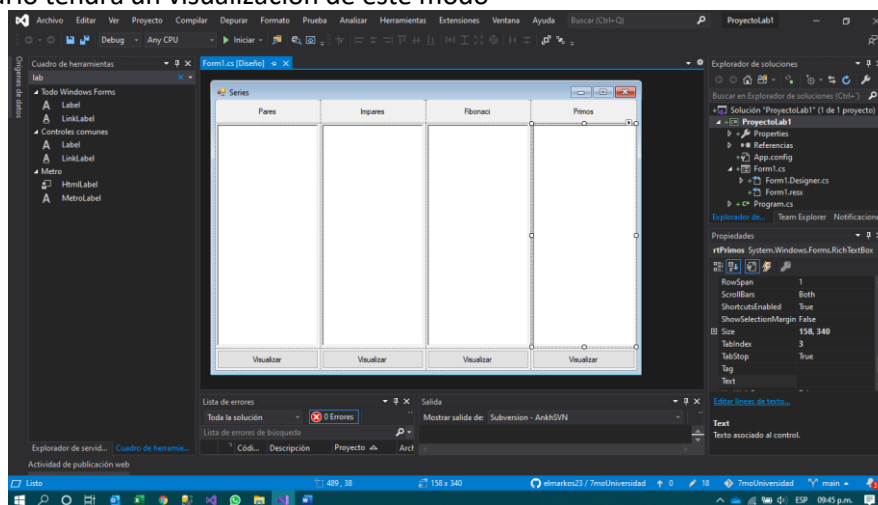
En la parte izquierda de Visual Studio tenemos los controles que podremos seleccionar o arrastrar a nuestro formulario, en este caso utilizaremos los siguientes controles:

El requerimiento es mostrar en pantalla las series pares, impares, fibonaci y números primos, entonces

Primero agregaremos un `TableLayoutPanel` que es un contenedor de controles a la misma le daremos la forma como se visualiza, seguido le agregamos las etiquetas (Label) para mostrar los datos agregaremos el control texto Plano (`richTextBox`) y finalmente agregar botones (`Button`); a todos los controles les agregaremos nombres en su propiedad `Name` y `Text` respectivamente al menos a los textos y botones.



El formulario tendrá un visualización de este modo



Ahora damos doble clic en el botón pares y agregamos el siguiente código para generar los 20 números pares.

```
private void btnPares_Click(object sender, EventArgs e)
{
    rtPares.Clear();
    for (int i = 0; i <= 20; i++)
        rtPares.AppendText("\n" + i * 2);
}
```

Seguido de igual forma ingresamos al botón Impares y agregamos el siguiente código

```
1 referencia | 0 cambios | 0 autores, 0 cambios
private void btnImpares_Click(object sender, EventArgs e)
{
    rtImpares.Clear();
    for (int i = 0; i <= 20; i++)
        rtImpares.AppendText("\n" + (2 * i + 1));
}
```

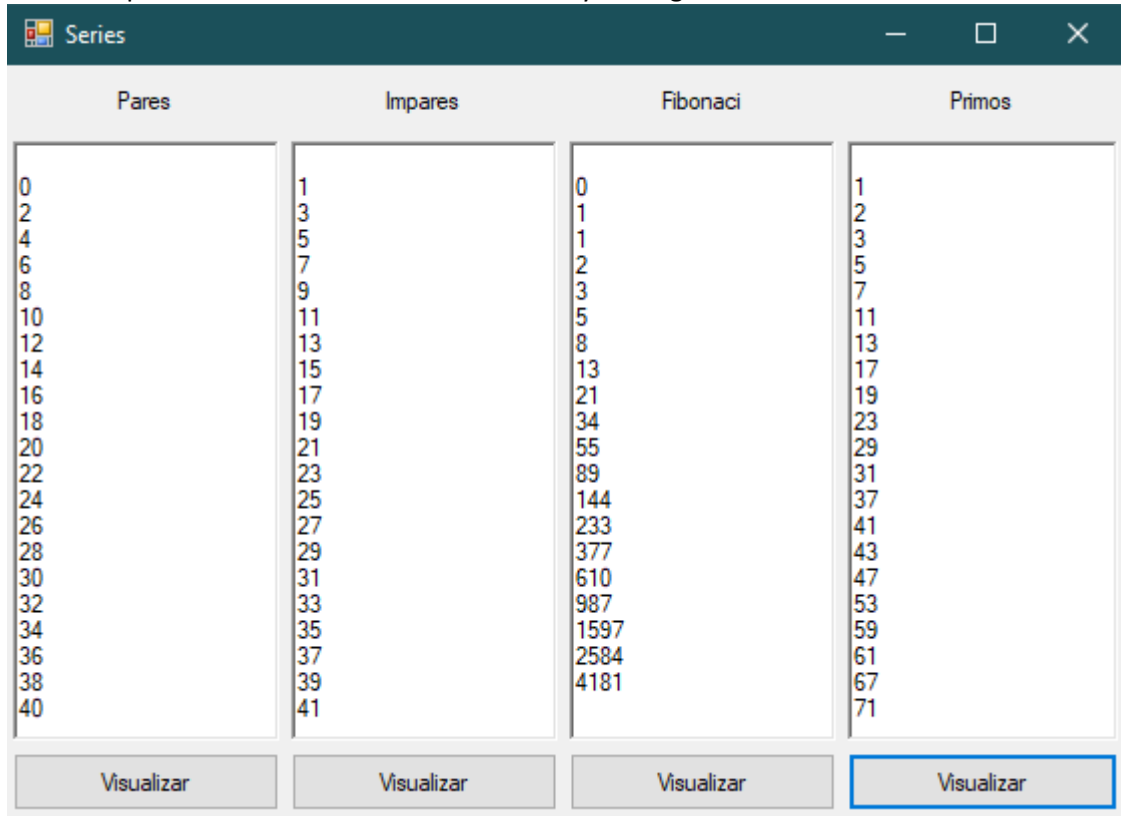
Ahora agregamos el siguiente código en el botón de visualizar la serie fibonacci

```
private void btnFibonacci_Click(object sender, EventArgs e)
{
    rtFibonacci.Clear();
    int f = 0; int t1 = 1;
    int t2;
    for (int i = 1; i <= 20; i++)
    {
        t2 = f;
        f = t1 + f;
        t1 = t2;
        rtFibonacci.AppendText("\n" + t1);
    }
}
```

Finalmente agregamos el siguiente código en el botón primos

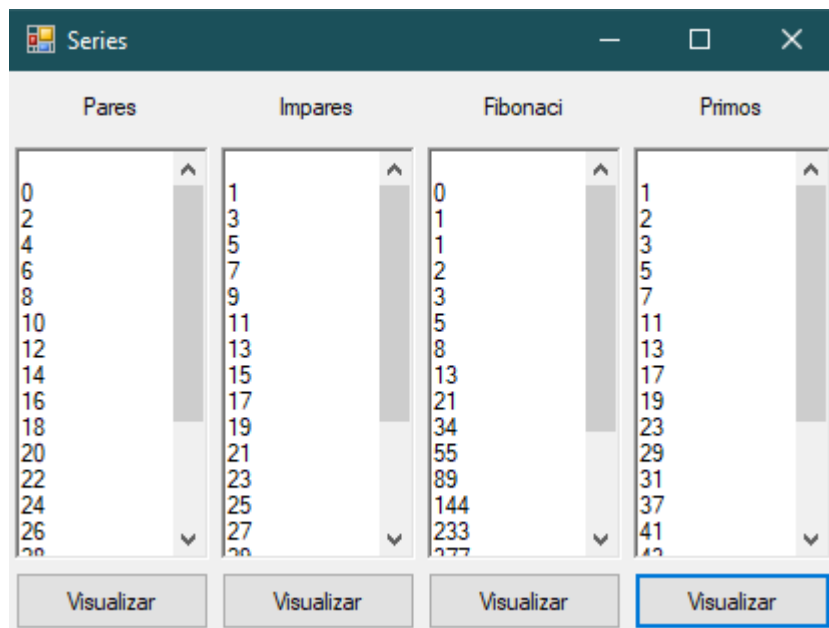
```
private void btnPrimos_Click(object sender, EventArgs e)
{
    rtPrimos.Clear();
    int contador = 0, num = 1, auxiliar = 0;
    do
    {
        for (int i = 1; i <= num; i++)
            if (num % i == 0) contador = contador + 1;
        if (contador <= 2)
        {
            rtPrimos.AppendText("\n" + num); auxiliar++;
        }
        num++;
        contador = 0;
    }
    while (auxiliar <= 20);
}
```

Ya finalizado podemos ejecutar el programa, claro primero debemos compilar. Ya cuando este abierto se preciona clic en los botones visualizar y se cargará la información.



Pares	Impares	Fibonaci	Primos
0	1	0	1
2	3	1	2
4	5	1	3
6	7	2	5
8	9	3	7
10	11	5	11
12	13	8	13
14	15	13	17
16	17	21	19
18	19	34	23
20	21	55	29
22	23	89	31
24	25	144	37
26	27	233	41
28	29	377	43
30	31	610	47
32	33	987	53
34	35	1597	59
36	37	2584	61
38	39	4181	67
40	41		71

Opcional. Si podemos ver que la pantalla se renderizo al tamaño deseado más pequeño, así que nuestros controles se reajustan y continúan mostrando información.



Pares	Impares	Fibonaci	Primos
0	1	0	1
2	3	1	2
4	5	1	3
6	7	2	5
8	9	3	7
10	11	5	11
12	13	8	13
14	15	13	17
16	17	21	19
18	19	34	23
20	21	55	29
22	23	89	31
24	25	144	37
26	27	233	41
28	29	377	43

5. CONCLUSIONES:

- En Visual Studio se puede realizar aplicaciones multiplataforma
- En Visual Studio Soporta varios lenguajes de programación.
- Con C# es un lenguaje super robusto.
- Con C# se puede crear aplicaciones de diferentes tipos.

6. RECOMENDACIONES:

- Con la librería de Xamarin.Forms se puede crear aplicaciones móviles nativas para Android y los.
- La integración de librerías y componentes open source.
- El lenguaje base como lo es C# y la integración de frameworks lo hace un potencial para crear aplicaciones.
- Las aplicaciones de consola, desktop, web, servicios web, servicios windos, y móvil se pueden crear desde el lenguaje de programación c#

7. BIBLIOGRAFIA:

Libro electrónico: C Sharp (2000). C Sharp. Recuperado de
https://es.wikipedia.org/wiki/C_Sharp

Libro electrónico: Desarrollo de Aplicaciones .NET
<https://channel9.msdn.com/Series/CSharp-101/What-is-C/player>

Video: Que es C#? <https://channel9.msdn.com/Series/CSharp-101/What-is-C/player>

Libro electrónico: W, Bill. (2020). Novedades de C# 9.0. Recuperado de
<https://docs.microsoft.com/es-mx/dotnet/csharp/whats-new/csharp-9>

Libro electrónico: W, Bill. (2020). Introducción C# y .NET. Recuperado de
<https://docs.microsoft.com/es-mx/dotnet/csharp/getting-started/>