

Universidad Rafael Landívar  
Facultad de Ingeniería  
Ingeniería Química  
Introducción a la Programación – Sección 07  
Catedrático: Ing. Edwin Timoteo Chocoy Córdón

## **PROYECTO PRÁCTICO No. 01**

Amada Jimena García Cordero – 1092223  
María Alexandra Sánchez Vargas – 1055323

Guatemala, 16 de octubre de 2023

# ÍNDICE

INTRODUCCIÓN .....	i
ANÁLISIS.....	1
ENTRADAS.....	1
SALIDAS .....	2
PROCESOS .....	3
RESTRICCIONES .....	4
DISEÑO .....	5
DIAGRAMA DE FLUJO .....	5
CONCLUSIONES .....	6
RECOMENDACIONES .....	7
REFERENCIAS .....	8
LIBRERÍAS UTILIZADAS Y SU UTILIZACIÓN .....	8
REFERENCIAS BIBLIOGRÁFICAS .....	8
ANEXOS.....	9
MANUAL DEL USUARIO .....	9

## INTRODUCCIÓN

Para la clase de Introducción a la Programación, se realizó un proyecto el cual aborda la problemática del sistema de transporte público “Transmetro” de la municipalidad de Guatemala, donde se llegó a los 650,000 usuarios. Por lo anterior, se busca facilitar el proceso de la movilización en Guatemala, por lo que se propone implementar un sistema informático que ayude a los usuarios del transmetro a comprar boletos para sus futuros viajes y proporcionar al usuario información vital para su travesía, para así llevar un sistema de control más planificado en las rutas.

El programa se encuentra diseñado para que cada que un usuario desee comprar un boleto para el transmetro, el usuario deba ingresar la estación de origen, la estación de destino, la fecha en la que se desee realizar el viaje, edad del pasajero, situación del pasajero (si se encuentra embarazada o no) y si el pasajero viaja con un niño de 3 años o menor. Gracias a este sistema, los usuarios recibirán una ruta detallada para llegar a su destino, conocerán el precio a pagar por su boleto (con descuento si son estudiantes o la exención de tarifas si son mujeres embarazadas o padres de niños menores a 3 años), al igual que el tiempo estimado de su viaje, la cual se basa en la cantidad de kilómetros que toma la ruta y a la velocidad en la que se encuentra el transmetro, la cual es 40 km/h. Una vez completado el proceso, se imprimirán los datos del boleto para su referencia.

Pero el programa no termina ahí, pues le brinda al usuario la oportunidad de planificar otro viaje, donde la estación inicial de este nuevo viaje es la estación final del primer viaje. Dependiendo si el usuario desee o no otro viaje. El programa concluye mostrando todas las etapas de la travesía, un resumen del tiempo total del viaje y el gasto acumulado por las rutas.

Para la creación del programa para los usuarios del transmetro, se utilizó Visual Studio, el cual es un entorno de desarrollo integrado que permite a los programadores crear aplicaciones para distintas plataformas. Este software ofrece diversas herramientas para facilitar el proceso de desarrollo, depuración de código, integración con diversos sistemas, pruebas automatizadas, escrituras de códigos, entre otras herramientas

# ANÁLISIS

## ENTRADAS

Imagen No. 01

```
0 referencias
static void Main(string[] args)
{
    InicializarRutas();

    Console.WriteLine("¡Bienvenido al sistema de compra de boletos del Trasmetro!");
    decimal totalGastado = 0;
    int tiempoTotalViaje = 0;
    int estacionDestino;
    int estacionPartida;
    string rutaKey;
    Console.ReadKey();
}
```

Esta es una entrada del programa ya que en esta sección se declaran las variables con las que se estará trabajando y se usarán en la estructura del código.

Imagen No. 02

```
// Ingreso de datos
Console.Clear();
Console.Write("Por favor, ingresa la estación de partida (código): ");
estacionPartida = int.Parse(Console.ReadLine());
string nombreEstacionPartida = estaciones.ContainsKey(estacionPartida) ? estaciones[estacionPartida] : "Desconocida";

Console.Write("Ahora, ingresa la estación de destino (código): ");
estacionDestino = int.Parse(Console.ReadLine());
string nombreEstacionDestino = estaciones.ContainsKey(estacionDestino) ? estaciones[estacionDestino] : "Desconocida";

rutaKey = GenerarRutaKey(estacionPartida, estacionDestino);
if (rutas.ContainsKey(rutaKey))
{
    Console.WriteLine("\nIngresa la fecha de tu viaje (AAAA-MM-DD): ");
    DateTime fechaViaje = DateTime.Parse(Console.ReadLine());

    Console.Write("Por favor, dinos tu edad: ");
    int edad = int.Parse(Console.ReadLine());

    Console.Write("¿Estás embarazada? (Sí/No): ");
    bool embarazada = Console.ReadLine().Equals("Si", StringComparison.OrdinalIgnoreCase);

    Console.Write("¿Viajarás con un niño menor de 3 años? (Sí/No): ");
    bool conNinoMenor = Console.ReadLine().Equals("Si", StringComparison.OrdinalIgnoreCase);

    decimal precioBoleto = CalcularPrecioBoleto(distancias[rutaKey], edad, embarazada, conNinoMenor);
    int tiempoViajeEstimado = CalcularTiempoViaje(distancias[rutaKey]);
}
```

Esta sección también es considerada una entrada porque aquí es donde se le pide al usuario ingresar los datos necesarios para que pueda comprar el boleto de Trasmetro y el programa pueda determinar el tiempo aproximado del viaje y el total a pagar.

Imagen No. 03

```
Console.ReadKey();
Console.Clear();
Console.WriteLine("\n¿Deseas realizar otro viaje? (Sí/No): ");
string continuarViaje = Console.ReadLine();
```

Esta sección del programa es una entrada ya que el usuario puede ingresar si desea realizar otro viaje, de ser así el programa vuelve a iniciar tomando como estación de partida la estación final del primer viaje.

## SALIDAS

**Imagen No. 04**

```
// Revelación de datos de boleto a comprar
Console.Clear();
Console.WriteLine($"Ruta de partida: {nombreEstacionPartida}");
Console.WriteLine($"Ruta de destino: {nombreEstacionDestino}");
Console.WriteLine("Precio del boleto: " + precioBoleto.ToString("C"));
Console.WriteLine($"Tiempo estimado de viaje: {tiempoViajeEstimado} minutos");

totalGastado += precioBoleto;
tiempoTotalViaje += tiempoViajeEstimado;
```

Esta es una salida ya que en esta parte del programa es donde se le muestra al usuario los datos que este ingresó con anterioridad, antes de imprimir los datos finales del boleto.

**Imagen No. 05**

```
// Boleto comprado
Console.Clear();
Console.WriteLine($"Ruta: {rutasNombres[rutaKey]}");
Console.WriteLine($"Tiempo total viajando: {tiempoTotalViaje} minutos");
Console.WriteLine($"Total gastado en boletos: {totalGastado.ToString("C")}");
Console.ReadKey();
```

En esta sección es donde se muestra al usuario el resultado final al realizar por completo el proceso de la compra de los boletos. Aquí se muestra el valor total a pagar y la duración promedio del viaje.

## PROCESOS

Imagen No. 06

```
// Rutas posibles
1 referencia
static void InicializarRutas()
{
    AgregarRuta(51, 61);
    AgregarRuta(51, 71);
    AgregarRuta(71, 82);
    AgregarRuta(61, 51);
    AgregarRuta(82, 51);
}

5 referencias
static void AgregarRuta(int estacionPartida, int estacionDestino)
{
    string rutaKey = GenerarRutaKey(estacionPartida, estacionDestino);
    rutas.Add(rutaKey, distancias[rutaKey]);
}

2 referencias
static string GenerarRutaKey(int estacionPartida, int estacionDestino)
{
    return $"{estacionPartida}-{estacionDestino}";
}

1 referencia
static decimal CalcularPrecioBoleto(int distancia, int edad, bool embarazada, bool conNinoMenor)
{
    decimal precioBase = distancia <= 10 ? distancia * 0.07m : 10 * 0.07m + (distancia - 10) * 0.02m;
    switch (edad)
    {
        case int n when (n >= 15 && n <= 25):
            return embarazada || conNinoMenor ? 0 : precioBase * 0.79m;
        default:
            return embarazada || conNinoMenor ? 0 : precioBase;
    }
}

2 referencias
static string GenerarRutaKey(int estacionPartida, int estacionDestino)
{
    return $"{estacionPartida}-{estacionDestino}";
}

1 referencia
static decimal CalcularPrecioBoleto(int distancia, int edad, bool embarazada, bool conNinoMenor)
{
    decimal precioBase = distancia <= 10 ? distancia * 0.07m : 10 * 0.07m + (distancia - 10) * 0.02m;
    switch (edad)
    {
        case int n when (n >= 15 && n <= 25):
            return embarazada || conNinoMenor ? 0 : precioBase * 0.79m;
        default:
            return embarazada || conNinoMenor ? 0 : precioBase;
    }
}

1 referencia
static int CalcularTiempoViaje(int distancia)
{
    // Calcula el tiempo en minutos sobre los 40km de la velocidad del bus
    return (int)(distancia * 60 / 40);
}
```

Este es el proceso de todo el programa, en esta sección del código se calcula el tiempo del viaje en base a la estación de partida y la estación de destino, también se calcula el total a pagar dependiendo de los kilómetros recorridos por la ruta tomada y también se aplican los descuentos si el usuario viaja con niños o si está embarazada.

## RESTRICCIONES

Imagen No. 07

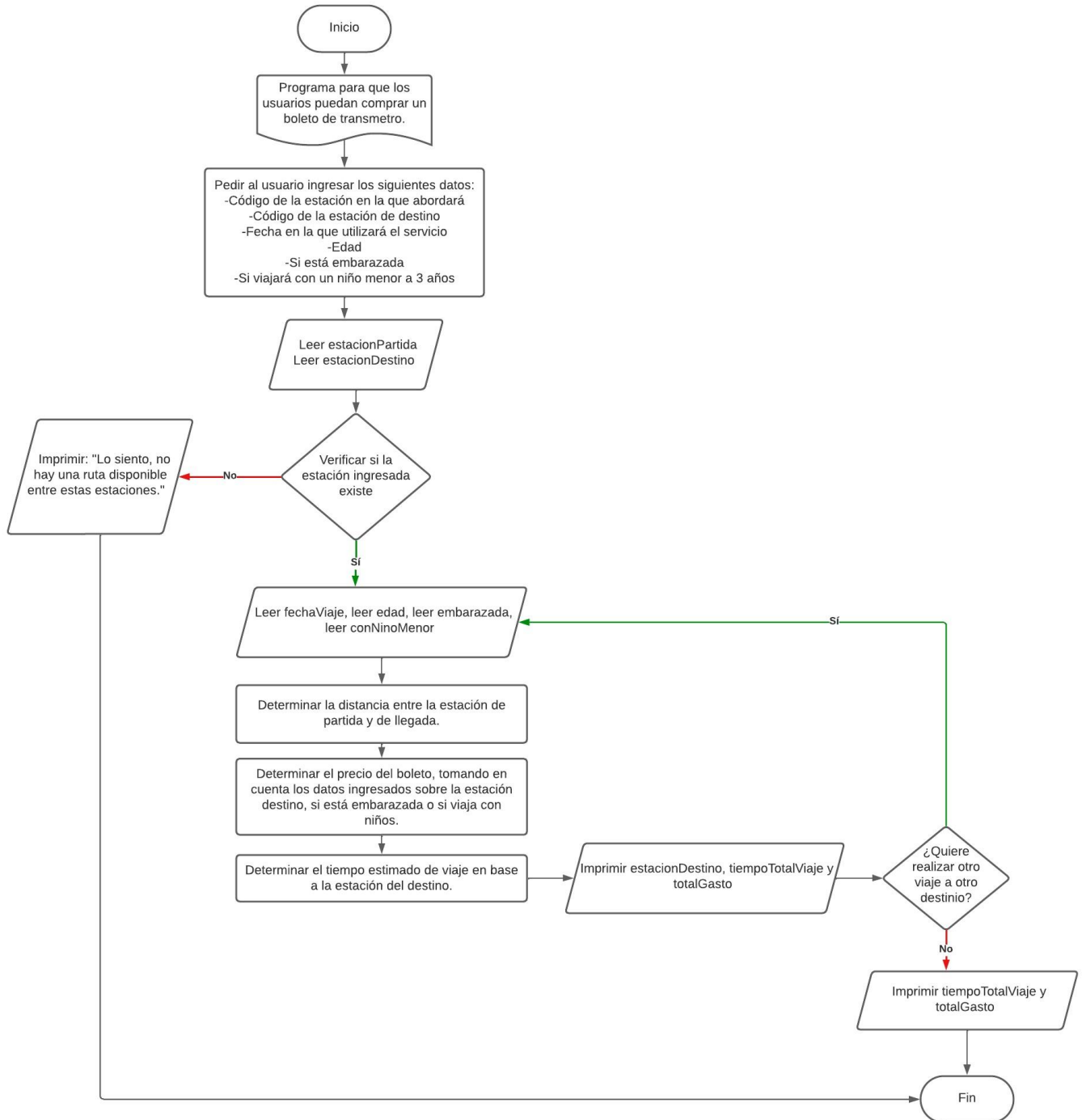
```
103  
104  
105  
106     else  
107     {  
108         Console.WriteLine("Lo siento, no hay una ruta disponible entre estas estaciones.");  
109     }  
110     Console.ReadKey();  
111 }  
112
```

Esta sección es una restricción ya que si el usuario ingresa una ruta que no se encuentra en el programa este no podrá avanzar y se mostrará el siguiente mensaje: " Lo siento, no hay una ruta disponible entre estas estaciones". El mismo se mostrará hasta que el usuario ingrese un código válido para fijar una estación de destino.

# DISEÑO

## DIAGRAMA DE FLUJO

**Diagrama No. 01**  
Proceso del Programa



Fuente: Elaboración Propia (2023)



## CONCLUSIONES

1. Para la creación del programa que genera un boleto de pasaje para realizar un viaje en el transmetro de Guatemala, este solicitó al usuario ciertos datos los cuales se contemplaron como variables; decimal para el total gastado por el uso de números decimales, int para el tiempo total en minutos y las estaciones de partida y destino, y la variable string para la ruta a tomar por el pasajero.
2. Cuando se creó el programa, se utilizó el ciclo while con el objetivo de poder repetir el proceso si el usuario deseara realizar otro viaje, mientras que los minutos y el precio se iban acumulando, dependiendo de la extensión de sus rutas.
3. Gracias a la elaboración del programa, la municipalidad de Guatemala posee un sistema que no solamente simplifica la compra de boletos para realizar viajes por medio del transmetro de una forma eficaz, sino que también transforma la experiencia en que las personas se movilizan.
4. Un proyecto de programación es esencial en la era tecnológica actual, ya que permite crear aplicaciones, software y sistemas que mejoran la eficiencia y calidad de vida. Fomenta la innovación, promueve el aprendizaje y la colaboración, y contribuye al avance tecnológico y al progreso de la sociedad.

## RECOMENDACIONES

1. Antes de comenzar a escribir el programa, es fundamental tomarse un tiempo para planificar y diseñar cómo será. Esto significa pensar detenidamente en lo que se quiere lograr con el programa y cómo se debería estructurar. Esto ayuda a evitar confusiones más adelante y asegura que el resultado final sea lo que se esperaba desde el principio.
2. Es fundamental escribir código de programación de forma clara y organizada para asegurar que otras personas puedan entenderlo fácilmente. Para lograrlo, se deben seguir reglas simples de nombrado de variables y funciones, dividir el código en partes lógicas más pequeñas y explicar en comentarios las secciones más importantes. Evitar repetir trozos de código y mantenerlo limpio ayudará a que sea más fácil de leer y modificar en el futuro.
3. Es esencial probar el programa para asegurarse de que funcione como se espera y cumpla con lo que se planeó. Realizar diferentes pruebas para verificar que cada parte del programa funcione correctamente es importante. Además, es fundamental corregir cualquier error o problema que se encuentre durante estas pruebas.
4. Cuando sea momento de pedirle al usuario que ingrese algún tipo de dato, especificarle al usuario como se desea que se ingrese el dato exacto. Existen distintos tipos de datos, por lo que cada uno conlleva una variable diferente, por lo que el sistema puede dejar de funcionar si se ingresa el dato erróneo.

## REFERENCIAS

### LIBRERÍAS UTILIZADAS Y SU UTILIZACIÓN

**using System.Collections.Generic:** Contiene interfaces y clases que definen colecciones genéricas, lo que permite que los usuarios creen colecciones fuertemente tipadas para proporcionar una mayor seguridad de tipos y un rendimiento mejor que los de las colecciones no genéricas fuertemente tipadas.

### REFERENCIAS BIBLIOGRÁFICAS

EDraw. (2023). *Cómo crear un Diagrama de Flujo*. Wondershare.

<https://www.edrawsoft.com/es/create-programming-flowchart.html>

apr. (s.f.) *Tipos de variables en Visual Basic*.

[https://www.aprenderaprogramar.com/index.php?option=com\\_content&view=article&id=140:tipos-de-variables-en-visual-basic-integer-single-double-string-object-boolean-etc-ejemplos-cu00308a&catid=37&Itemid=61](https://www.aprenderaprogramar.com/index.php?option=com_content&view=article&id=140:tipos-de-variables-en-visual-basic-integer-single-double-string-object-boolean-etc-ejemplos-cu00308a&catid=37&Itemid=61)

DevCamp. (s.f.). *Qué es una librería en programación*. <https://devcamp.es/que-es-libreria-programacion/>

Armetrics. (s.f.). *Qué es Visual Studio*. <https://www.armetrics.com/glosario-digital/visual-studio>

## ANEXOS

### MANUAL DEL USUARIO

1. El usuario debe ingresar al programa en el caso que quiera comprar un boleto para realizar un viaje en el transmetro por la Ciudad de Guatemala.
2. El usuario debe conocer los códigos designados para cada estación que tiene el transmetro, las cuales son las siguientes:

Código	Nombre estación
51	Estación Javier
61	Estación Trébol
71	Estación Don Bosco
82	Estación Plaza Municipal

3. El programa le pide al usuario que ingrese la siguiente información:
  - Código de la estación en la que desee abordar
  - Código de la estación de destino
  - Fecha en la que se realizará el viaje
  - Edad del usuario
  - Indicar si el usuario está embarazado
  - Indicar si el usuario viaja con un niño menor a 3 años
4. Con la información dada anteriormente por el usuario, el programa le enseña al usuario su ruta, un estimado del precio del boleto y el tiempo estimado del viaje.
5. Seguidamente, el programa le pregunta al usuario si desea realizar otro viaje, en el cual su destino inicial sería su destino final del primer viaje.
6. Si el usuario no desea realizar otro viaje, ingrese “no”, seguidamente se le mostrará su boleto para el transmetro donde se demuestra su ruta final, su tiempo de viaje y su total gastado en el viaje.
7. Si el usuario desea realizar otro viaje, ingrese “sí”, seguidamente se le pedirá al usuario que ingrese el código de su nueva estación de destino.
8. Por último, se le mostrará al usuario su boleto para el transmetro donde se demuestra su ruta final, su tiempo de viaje y su total gastado en el viaje.

**\*Si el usuario se encuentra en estado de embarazo o viaja con un niño menor a 3 años, el boleto para el transmetro le sale gratis. Los usuarios entre 15 y 25 años son considerados como estudiantes, por lo que obtienen un 21% de descuento en el precio total a pagar por el boleto para el transmetro\***