

## Examen: Diseño de Sistema de Control de Migración en C++ ()

### Instrucciones Generales:

Este examen tiene como objetivo evaluar tu capacidad para diseñar un sistema de control de migración utilizando C++, únicamente utilizando funciones y métodos. El sistema gestionará el ingreso y salida de viajeros en un país. Deberás manejar la información utilizando arreglos o matrices y proporcionar un menú interactivo para las operaciones principales.

Debes analizar lo que involucra el control de migración de viajeros que salen y entran al país, crear estructuras al menos 5 estructura por cada elemento por ejemplo (

**struct Viajero {**

int id;

string nombre;

string direccion;

}

**struct Ingreso {**

int idViajero;

string fecha salidad;

string LugarIngreso;

}

### Requisitos del Sistema:

#### 1. Estructura de Datos:

- **Arreglos o Matrices:**

- Utiliza arreglos o matrices para almacenar la información de los viajeros y los registros de entrada y salida.
- Por ejemplo, podrías usar un arreglo de estructuras para manejar los datos de los viajeros.

#### 2. Menú Principal:

- **Interfaz de Usuario:**

- Diseña un menú interactivo en la consola que permita al usuario acceder a las diferentes funcionalidades del sistema

- Asegúrate de que el menú sea claro y fácil de usar, permitiendo al usuario navegar entre las opciones.

### 3. Operaciones y Funcionalidades:

- **Ingreso Viajero:**
  - Implementa la funcionalidad para agregar nuevos viajeros al sistema, almacenando la información en el arreglo correspondiente.
- **Registrar ingreso o salida:**
- **Borrar registros:** eliminar o poner el blanco los datos del viajero
- **Consultar Información:**
  - Permite consultar la información de un viajero específico y mostrarla en la consola.
- **Generar Reportes:**
  - Crea una función que genere reportes y los muestre en la consola, como la lista de todos los viajeros o los registros de entradas y salidas.

### 4. Entrega:

- Debes entregar el código fuente completo del proyecto.

### Criterios de Evaluación:

- **Correctitud del Código:** Evaluación de la correcta implementación de las funciones.
- **Funcionalidad del Sistema:** Verificación de que todas las características solicitadas funcionen correctamente.
- **Calidad del Código:** Evaluación de la claridad, organización y buenas prácticas en el código fuente.
- **Interfaz de Usuario:** Evaluación de la usabilidad y diseño del menú.
- **Documentación:** Calidad y claridad de la documentación entregada.