

**UNIVERSIDAD DE SAN CARLOS DE GUATEMALA**  
**FACULTAD DE INGENIERÍA**  
**ESCUELA DE CIENCIAS Y SISTEMAS**  
**ESTRUCTURA DE DATOS**  
**ING. LUIS ESPINO**  
**AUX. ROBINSON PEREZ**  
**SECCION: "A"**

# **MANUAL**

# **TECNICO**

Nombre: Carlos Enrique Cante López  
Carnet: 201314448

## **INTRODUCCION**

La aplicación fue desarrollada en el lenguaje de programación C++ haciendo uso del IDE QT Creator en el sistema operativo Windows y generando las graficas con la aplicación Graphviz en su versión 2.38.

La Aplicación hace uso de estructuras de datos propias para la simulación de la lógica de un aeropuerto.

## **OBJETIVOS**

### **OBJETIVOS GENERALES:**

- Que la aplicación simule el funcionamiento de un aeropuerto

### **OBJETIVOS ESPECIFICOS:**

- Hacer uso de estructuras de datos propias
- Usar Graphviz para mostrar graficas de estructuras de datos
- Simular varias estructuras de datos con sus respectivos métodos.

## **DESCRIPCIÓN DEL PROBLEMA**

Se desea realizar un simulador de aeropuerto por medio de estructuras de datos propias haciendo que estas interactúen entre si de modo correcto y mostrando el estado actual de dichas estructuras en una imagen generada por la aplicación Graphviz.

## C++

C++ es un lenguaje de programación diseñado a mediados de los años 1980 por Bjarne Stroustrup. La intención de su creación fue el extender al lenguaje de programación C mecanismos que permiten la manipulación de objetos. En ese sentido, desde el punto de vista de los lenguajes orientados a objetos, el C++ es un lenguaje híbrido.

Posteriormente se añadieron facilidades de programación genérica, que se sumaron a los paradigmas de programación estructurada y programación orientada a objetos. Por esto se suele decir que el C++ es un lenguaje de programación multiparadigma.

Actualmente existe un estándar, denominado ISO C++, al que se han adherido la mayoría de los fabricantes de compiladores más modernos. Existen también algunos intérpretes, tales como ROOT.

Una particularidad del C++ es la posibilidad de redefinir los operadores, y de poder crear nuevos tipos que se comporten como tipos fundamentales.

El nombre "C++" fue propuesto por Rick Mascitti en el año 1983, cuando el lenguaje fue utilizado por primera vez fuera de un laboratorio científico. Antes se había usado el nombre "C con clases". En C++, la expresión "C++" significa "incremento de C" y se refiere a que C++ es una extensión de C.

### **ESTRUCTURAS IMPLEMENTADA:**

- Aviones que aterrizan: COLA DOBLEMENTE ENLAZADA
- Escritorios: LISTA DOBLEMENTE ENLAZADA ORDENADA
- Cola de Persona: COLA SIMPLE
- Documentos: PILA SIMPLE
- Estaciones de Mantenimiento: LISTA SIMPLE
- Maletas: LISTA DOBLEMENTE ENLAZADA CIRCULAR.

## STRUCT DE CADA NODO DEL SISTEMA

Avion:

```
struct NodoAvion
{
    int ID;
    int Turnos_D;
    int Turnos_M;
    int No_Pasajeros;

    NodoAvion* siguiente;
    NodoAvion* anterior;

    NodoAvion(int id, int turnosd, int turnosm, int nopasajeros);
};
```

Escritorio:

```
struct NodoEscritorio
{
    int numero;
    int PersonasEnCola;
    NodoPersona* PersonaActual;
    ColaPersonas* Cola;
    PilaDocumentos* Pila;

    NodoEscritorio* siguiente;
    NodoEscritorio* anterior;

    NodoEscritorio(int num);
    void AnadirDocumentos(int numero, int id);
    void SacarDocumentos(int numero);
};
```

Maleta:

```
struct NodoMaleta
{
    int id;

    NodoMaleta* siguiente;
    NodoMaleta* anterior;
    NodoMaleta(int id_);
};
```

### Estación de Mantenimiento:

```
struct NodoMantenimiento
{
    int numero;
    string estado;
    NodoAvion* AvionActual;
    NodoMantenimiento* siguiente;
    NodoMantenimiento(int numero_);
};
```

### Persona:

```
struct NodoPersona
{
    int IDPersona;
    int NoMaletas;
    int NoDocumentos;
    int NoTurnosR;

    NodoPersona* siguiente;

    NodoPersona(int id, int maletas, int doc, int turnosr);
};
```

### Documento:

```
struct NodoDocumento
{
    int numero;
    int IdPersona;
    NodoDocumento* siguiente;

    NodoDocumento(int n, int idp);
};
```



## STRUCT DE CADA ESTRUCTURA

Cola de Aviones:

```
struct ColaAvionesAterrizan
{
    NodoAvion* Primero;
    NodoAvion* Ultimo;
    int IDAvion = 1;

    ColaAvionesAterrizan();
    void IngresarAvion();
    NodoAvion* CrearAvion();
    NodoAvion* SacarAvion();
    string GenerarSubGrafo();
};
```

Cola de Personas:

```
struct ColaPersonas
{
    NodoPersona* Primero;
    NodoPersona* Ultimo;
    int IDPersona = 1;

    ColaPersonas();
    void IngresarPersona(ListaMaletas* l);
    void IngresarPersona2(NodoPersona* nueva);
    void CargarPasajeros(int Cantidad, ListaMaletas* l);
    NodoPersona* GenerarPersona();
    NodoPersona* SacarPersona();
    string GenerarSubGrafo(int NumeroCola);
    bool EstaVacia();
};
```

Lista de Escritorios:

```
struct ListaEscritorios
{
    NodoEscritorio* Primero;
    NodoEscritorio* Ultimo;
    int IdDocumentos;

    ListaEscritorios();

    void AgregarEscritorio(int id);
    void CargarEscritorios(int NoEscritorios);
    string GenerarSubgrafo();

    void CargarPersonas(ColaPersonas* colageneral);
    bool HayEspacio();

    void PasarTurno(ListaMaletas* l);

    QString EstadoActual();
};
```

## Lista de Maletas:

```
struct ListaMaletas
{
    int IdMaleta;

    NodoMaleta* Primero;
    NodoMaleta* Ultimo;

    ListaMaletas();
    void IngresarMaleta();
    void CargarMaletas(int NoMaletas);
    void SacarMaletas(int Cantidad);
    void PopMaleta();

    string GenerarSubGrafo();

    int CantidadDeMaletas();
};
```

## Estaciones de Mantenimiento:

```
struct ListaMantenimiento
{
    NodoMantenimiento* Primero;
    NodoMantenimiento* Ultimo;

    ColaAviones* ColaAv;
    ListaMantenimiento();
    void CargarAviones();
    void IngresarEstacion(int id);
    void CargarEstaciones(int n);
    string GenerarSubGrafo();
    bool HayEspacio();

    void PasarTurnoMantenimiento();

    QString EstadoActual();
};
```

## Pila de Documentos:

```
struct PilaDocumentos
{
    NodoDocumento* cima;
    PilaDocumentos();
    void PilaPush(int num, int id);
    void PilaPop();
    int Tamano();

    string GenerarSubGrafo(int id);
};
```