



**ESCUELA POLITÉCNICA NACIONAL
FACULTAD DE INGENIERÍA DE SISTEMAS**

PROGRAMACIÓN I

PROYECTO PRIMER BIMESTRE

GR-1SI

Grupo 5

Integrantes:

Paulette Cusichagua

Adrián Jara

Danny Ponce

Shamir Rivera

Larsson Umatambo

Diego Villagómez

Lunes, 09 de junio del 2025

PROYECTO PRIMER SEMESTRE

MAYO 2025

1. Diagrama de flujo

Opción 1:

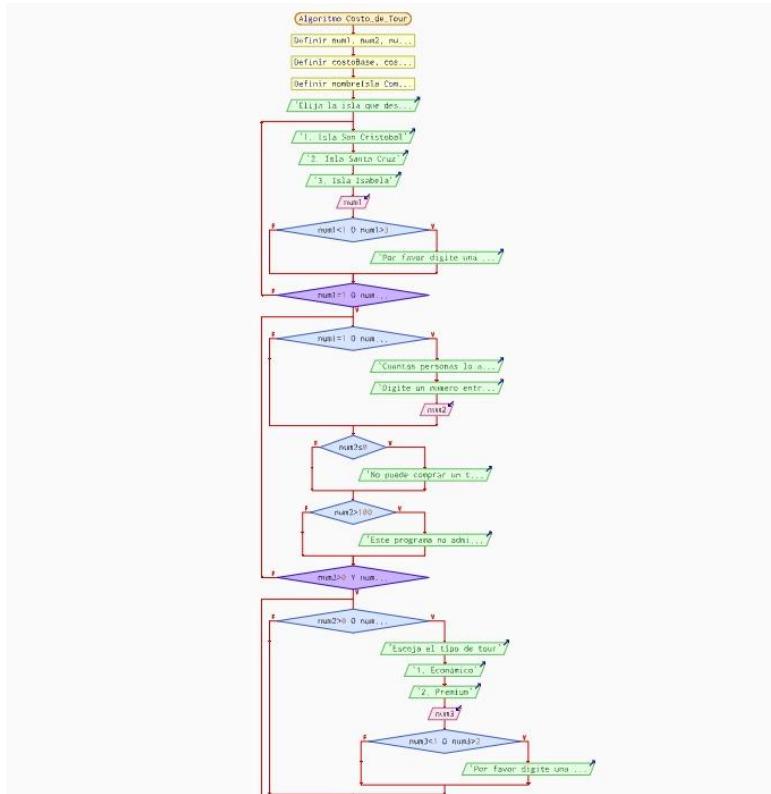


Figura [1]: Diagrama de flujo de la opción 1 (primera parte).

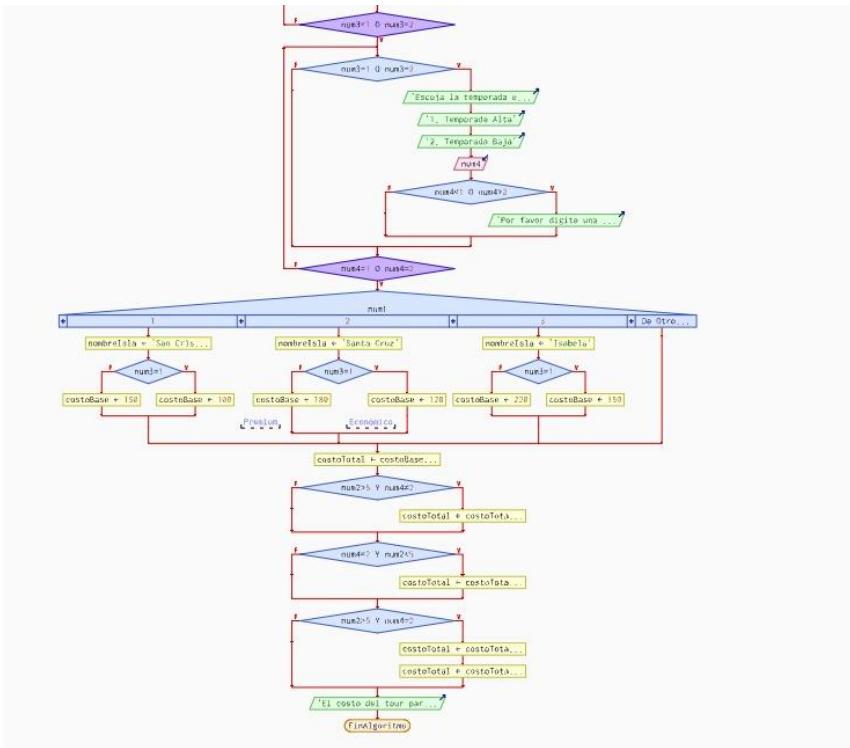


Figura [2]: Diagrama de flujo de la opción 1 (segunda parte).

Opción 2:

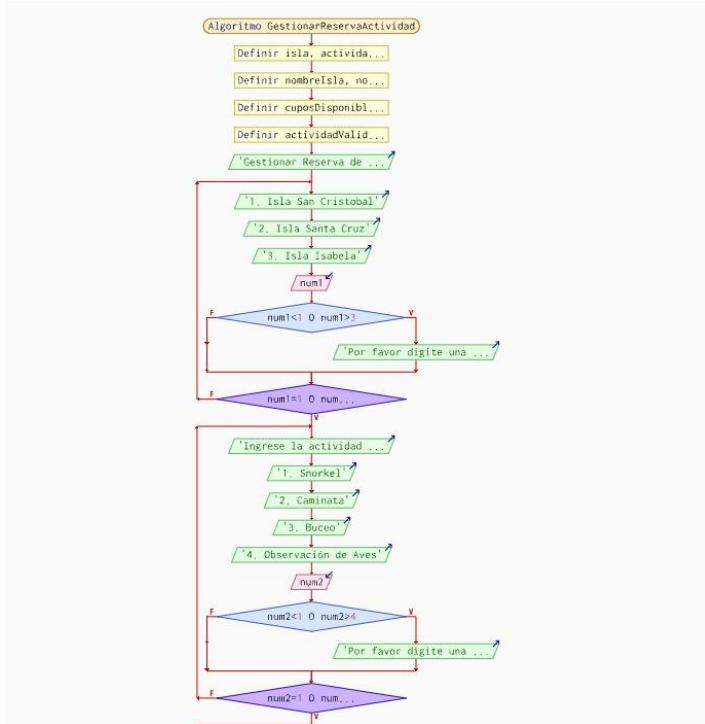


Figura [3]: Diagrama de flujo de la opción 2 (primera parte).

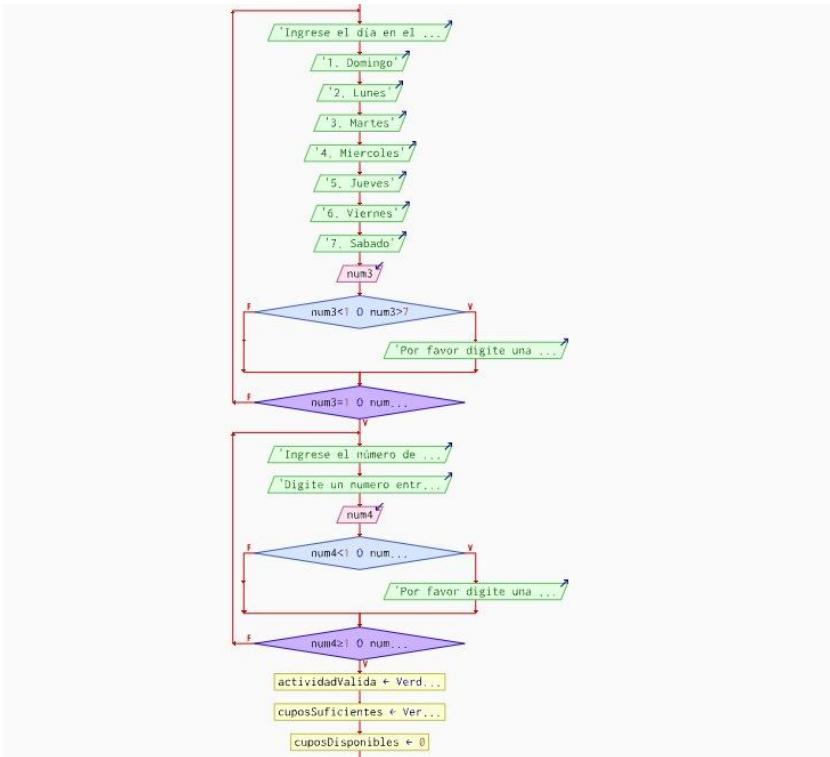


Figura [4]: Diagrama de flujo de la opción 2 (segunda parte).

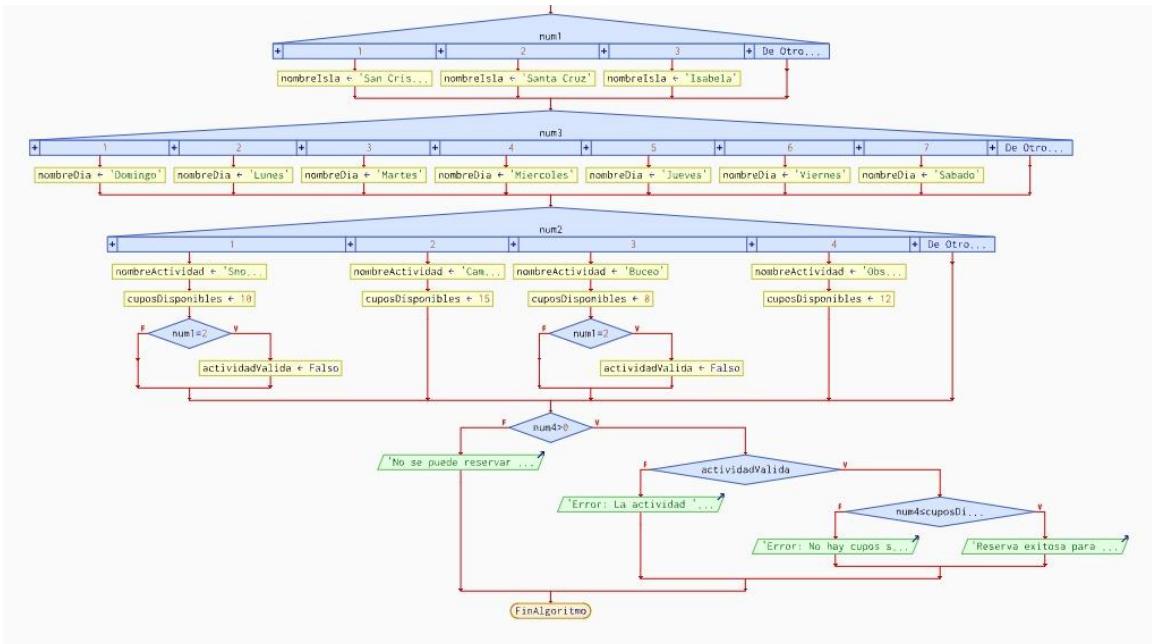


Figura [5]: Diagrama de flujo de la opción 2 (tercera parte).

Opción 3:

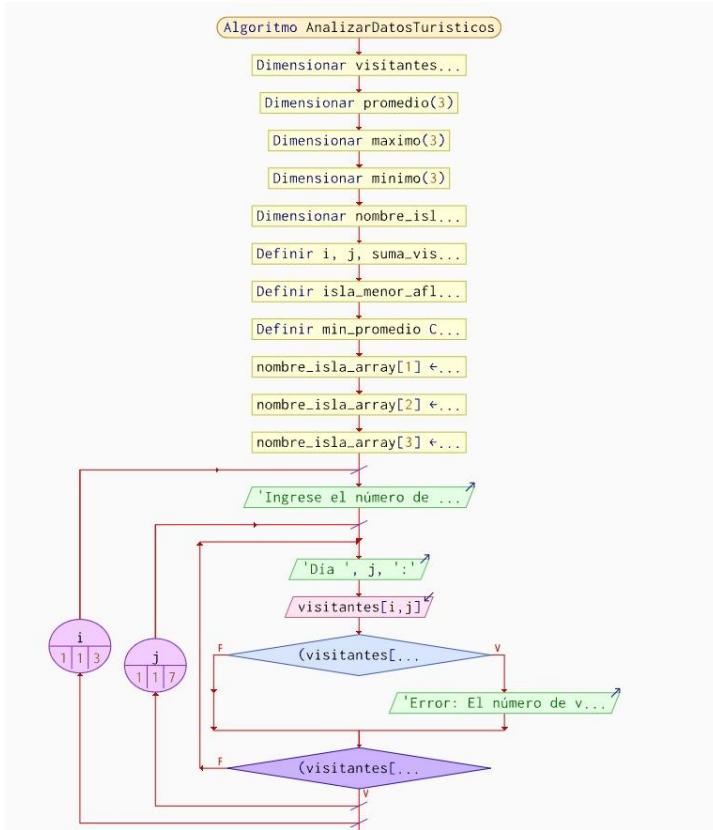


Figura [6]: Diagrama de flujo de la opción 3 (primera parte).

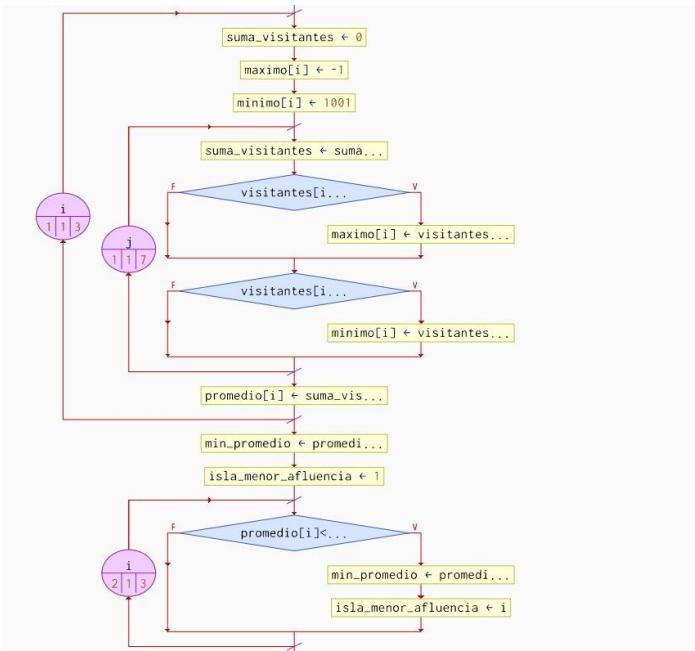


Figura [7]: Diagrama de flujo de la opción 3 (segunda parte).

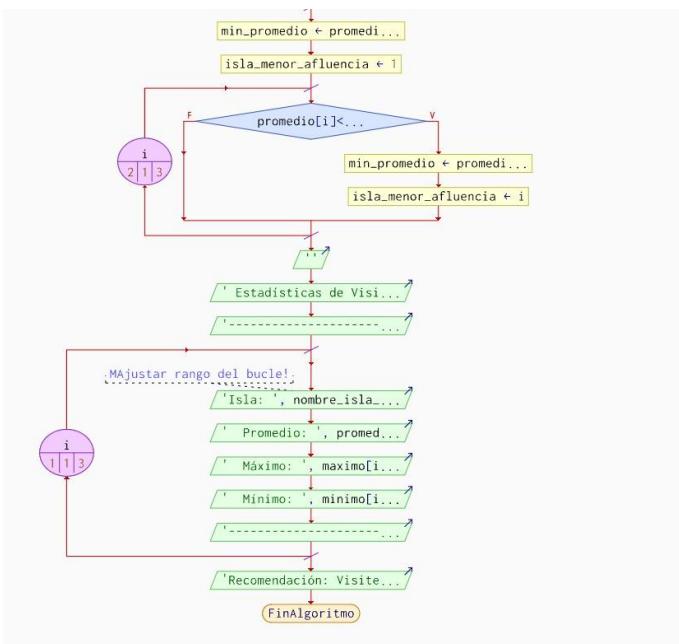


Figura [8]: Diagrama de flujo de la opción 3 (tercera parte).

Opción 4:

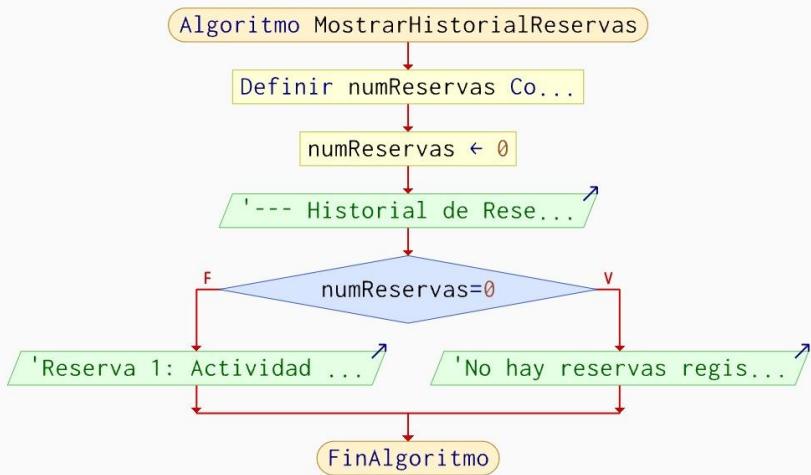


Figura [9]: Diagrama de flujo de la opción 4

Opción 5:

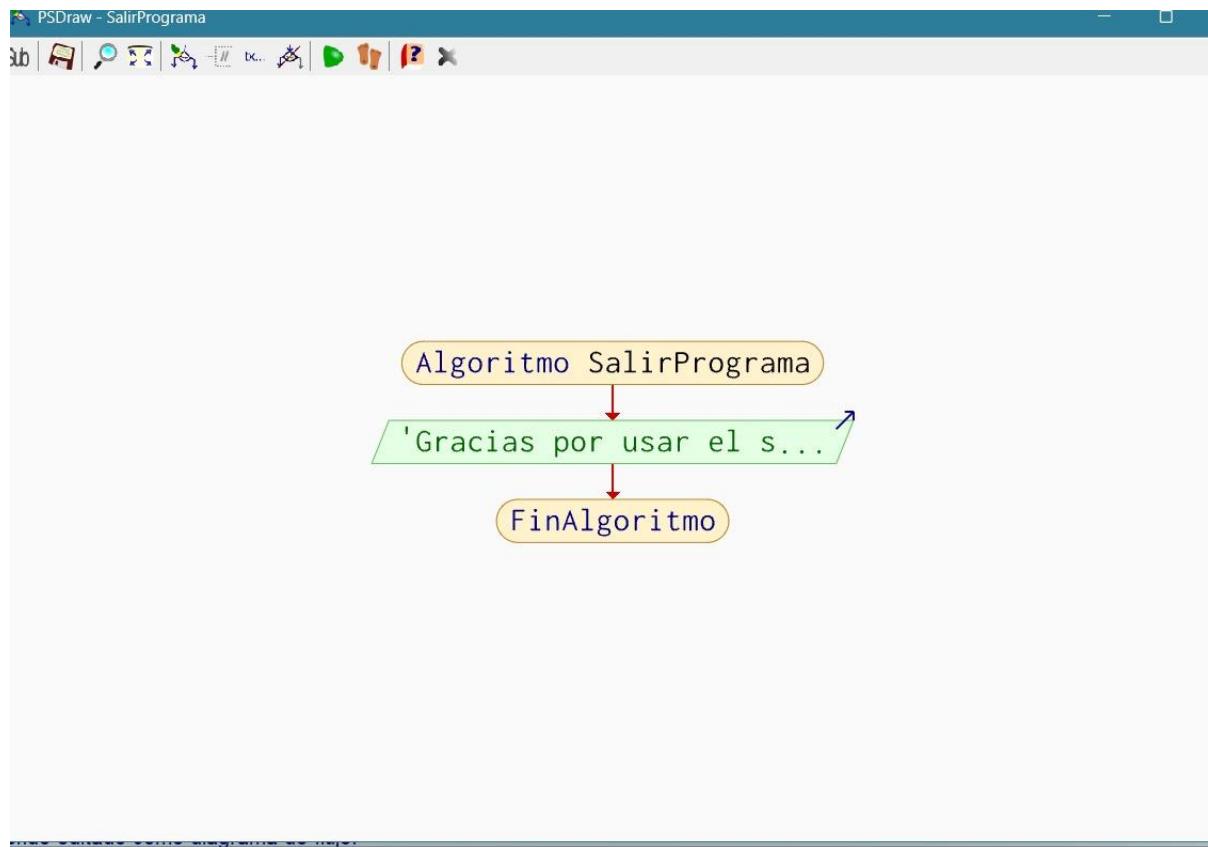


Figura [10]: Diagrama de flujo de la opción 5

2. Descripción de la construcción y funcionamiento de la aplicación

- 1) Empezamos abriendo Visual Studio Code, en el vamos a escribir las 5 opciones propuestas para el funcionamiento de nuestro sistema.
- 2) Para la primera opción empezamos escribiendo el código, en el vamos a limitar las opciones para elegir del usuario con el fin de evitar errores, el usuario podrá elegir la isla a la que quiere ir y para cuantas personas sería la reserva, de acuerdo con las opciones que elegía el usuario se mostrará el monto a pagar y si se cumple las condiciones para que le sea asignado un descuento.

```
// Esta función solo calcula el costo de un tour.  
void calcularCostoTour() {  
    std::cout << "n--- Calcular Costo de Tour ---\n";  
    int islaOpcion = obtenerEnterovalido("Ingrese la isla (1-San Cristóbal, 2-Santa Cruz, 3-Isabela): ", 1, 3);  
    int numPersonas = obtenerEnterovalido("Ingrese el numero de personas (1-100): ", 1, 100);  
    int tipoTour = obtenerEnterovalido("Ingrese el tipo de tour (1-Económico, 2-Premium): ", 1, 2);  
    int temporada = obtenerEnterovalido("Ingrese la temporada (1-Alta, 2-Baja): ", 1, 2);  
  
    double costoBase = 0.0;  
    std::string nombreIsla;  
  
    switch (islaOpcion) {  
        case 1:  
            nombreIsla = "San Cristóbal";  
            costoBase = (tipoTour == 1) ? COSTO_SC_ECO : COSTO_SC_PRE;  
            break;  
        case 2:  
            nombreIsla = "Santa Cruz";  
            costoBase = (tipoTour == 1) ? COSTO_STC_ECO : COSTO_STC_PRE;  
            break;  
        case 3:  
            nombreIsla = "Isabela";  
            costoBase = (tipoTour == 1) ? COSTO_ISA_ECO : COSTO_ISA_PRE;  
            break;  
    }  
  
    double costoTotal = costoBase * numPersonas;  
  
    if (numPersonas > 5) {  
        costoTotal *= 0.90;  
    }  
    if (temporada == 2) {  
        costoTotal *= 0.80;  
    }  
  
    std::cout << std::fixed << std::setprecision(2);  
    std::cout << "El costo del tour para " << numPersonas << " personas en Isla " << nombreIsla << " es de $" << costoTotal << ".\n";  
}
```

Figura [11]: Código opción 1

- 3) En la segunda opción, igualmente se delimitará las actividades que puede elegir el usuario. Se le solicitará al cliente que seleccione una actividad que quiera realizar, además del día en el cual quiere hacerlo. De acuerdo con la actividad seleccionada se delimitará el lugar en donde se puede realizar dicha actividad y cuantas personas es el máximo que pueden realizar esta actividad a la vez. Se concluirá con tres opciones:

Si la actividad es posible realizar aparecerá

- Reserva exitosa con los lugares y días en los cuales esta echa la reserva.

Si la actividad no es posible realizar aparecerá

- Reserva no exitosa, por falta de cupos de dicha actividad y lugar.
- Reserva no exitosa, porque la actividad solicitada no está dentro de los lugares designados.

```
// Encapsula la gestión de una reserva de actividad.
void gestionarReservaActividad() {
    std::cout << "\n--- Gestiónar Reserva de Actividad ---\n";
    int islaOpcion = obtenerEnterovalido("Ingrese la isla (1-San Cristóbal, 2-Santa Cruz, 3-Isabela): ", 1, 3);
    int actividadOpcion = obtenerEnterovalido("Ingrese la actividad (1-Snorkele, 2-Caminata, 3-Buceo, 4-Observación de aves): ", 1, 4);
    int dia = obtenerEnterovalido("Ingrese el día (1-7): ", 1, 7);
    int numPersonas = obtenerEnterovalido("Ingrese el numero de personas (1-50): ", 1, 50);

    std::string nombreIsla;
    switch (islaOpcion) {
        case 1: nombreIsla = "San Cristóbal"; break;
        case 2: nombreIsla = "Santa Cruz"; break;
        case 3: nombreIsla = "Isabela"; break;
    }

    std::string nombreActividad;
    int cupoMaximo = 0;
    bool actividadValida = true;

    switch (actividadOpcion) {
        case 1:
            nombreActividad = "Snorkel";
            if (islaOpcion == 2) {
                actividadValida = false;
            } else {
                cupoMaximo = CUPO_SNORKEL;
            }
            break;
        case 2:
            nombreActividad = "Caminata";
            cupoMaximo = CUPO_CAMINATA;
            break;
        case 3:
            nombreActividad = "Buceo";
            if (islaOpcion == 2) {
                actividadValida = false;
            } else {
                cupoMaximo = CUPO_BUCEO;
            }
            break;
        case 4:
            nombreActividad = "Observación de aves";
            cupoMaximo = CUPO_AVES;
            break;
    }

    if (!actividadValida) {
        std::cout << "Error: la actividad " << nombreActividad << " no esta permitida en la Isla " << nombreIsla << ".\n";
    } else if (numPersonas > cupoMaximo) {
        std::cout << "Error: El numero de personas (" << numPersonas << ") excede el cupo maximo para " << nombreActividad << "(" << cupoMaximo << ").\n";
    } else {
        historialReservas.push_back({nombreIsla, nombreActividad, dia, numPersonas});
    }
}

std::string nombreDia;
switch (dia) {
    case 1: nombreDia = "domingo"; break;
    case 2: nombreDia = "lunes"; break;
    case 3: nombreDia = "martes"; break;
    case 4: nombreDia = "miércoles"; break;
    case 5: nombreDia = "jueves"; break;
    case 6: nombreDia = "viernes"; break;
    case 7: nombreDia = "sábado"; break;
}

std::cout << "Reserva exitosa para actividad " << nombreActividad << " en Isla " << nombreIsla << " para el dia " << dia << " (" << nombreDia << ") para " << numPersonas << " personas.\n";
}
```

Figura [12]: Código opción 2

- 4) La tercera opción consistirá en realizar un estudio de cuantos turistas tendrá cada isla en los 7 días de la semana, se ingresará cuantas personas visitan la isla en el día, así durante toda la semana, con los datos se calculará un máximo, mínimo y promedio, al final de esto se nos informara cual es la isla que se nos recomienda visitar de acuerdo la numero de turistas.

```

148 // Funcion dedicada al analisis y presentacion de datos turisticos.
149 void analizarDatosTuristicos() {
150     std::cout << "\n--- Analizar Datos Turisticos ---\n";
151
152     std::string islas[] = {"San Cristobal", "Santa Cruz", "Isabela"};
153     const int NUM_ISLAS = 3;
154     int visitantes[3][NUM_ISLAS];
155
156     for (int i = 0; i < 3; ++i) {
157         std::cout << "Ingrese el numero de visitantes para Isla " << islas[i] << " (7 dias):\n";
158         for (int j = 0; j < NUM_ISLAS; ++j) {
159             visitantes[i][j] = obtenerEnteroValido("Dia " + std::to_string(j + 1) + " (0-1000): ", 0, 1000);
160         }
161     }
162
163     double promedioMinimo = std::numeric_limits<double>::max();
164     std::string islaMenorAfluencia;
165
166     std::cout << std::fixed << std::setprecision(2);
167
168     std::cout << "\nEstadisticas de Visitantes por Isla:\n";
169     std::cout << std::left << std::setw(15) << "Isla"
170         << std::setw(15) << "Promedio"
171         << std::setw(10) << "Maximo"
172         << std::setw(10) << "Minimo" << "\n";
173     std::cout << "-----\n";
174
175     for (int i = 0; i < 3; ++i) {
176         long long suma = 0;
177         int maxVisitantes = 0;
178         int minVisitantes = 1001;
179
180         for (int j = 0; j < NUM_ISLAS; ++j) {
181             suma += visitantes[i][j];
182             if (visitantes[i][j] > maxVisitantes) {
183                 maxVisitantes = visitantes[i][j];
184             }
185             if (visitantes[i][j] < minVisitantes) {
186                 minVisitantes = visitantes[i][j];
187             }
188         }
189
190         double promedio = static_cast<double>(suma) / NUM_ISLAS;
191
192         std::cout << std::left << std::setw(15) << islas[i]
193             << std::setw(15) << promedio
194             << std::setw(10) << maxVisitantes
195             << std::setw(10) << minVisitantes << "\n";
196
197         if (promedio < promedioMinimo) {
198             promedioMinimo = promedio;
199             islaMenorAfluencia = islas[i];
200         }
201     }
202     std::cout << "-----\n";
203     std::cout << "Recomendacion: Visite la Isla " << islaMenorAfluencia << " (es la que, en promedio, tiene menor afluencia).\n";
204 }

```

Figura [13]: Código opción 3

- 5) Nuestra cuarta opción realizará un historial donde esta almacenados todos los registros anteriores y en caso de no existir ninguno se mostrará que no existen registros anteriores, caso contrario el programa nos mostrará la reserva que incluirá, el día, la actividad realizada y en que isla se hizo.

```

// Encargada de mostrar el historial de reservas.
void mostrarHistorialReservas() {
    std::cout << "\n--- Historial de Reservas ---\n";
    if (historialReservas.empty()) {
        std::cout << "No hay reservas registradas.\n";
    } else {
        for (size_t i = 0; i < historialReservas.size(); ++i) {
            const auto& r = historialReservas[i];
            std::string nombreDia;
            switch (r.dia) {
                case 1: nombreDia = "domingo"; break;
                case 2: nombreDia = "lunes"; break;
                case 3: nombreDia = "martes"; break;
                case 4: nombreDia = "miercoles"; break;
                case 5: nombreDia = "jueves"; break;
                case 6: nombreDia = "viernes"; break;
                case 7: nombreDia = "sabado"; break;
            }
            std::cout << "Reserva " << (i + 1) << ": Actividad de " << r.actividad
                << " en Isla " << r.isla << " para el dia " << r.dia
                << " (" << nombreDia << ") para " << r.numPersonas << " personas.\n";
        }
    }
}

```

Figura [14]: Código opción 4

- 6) Finalmente, la quita opción será un agradecimiento al usuario por hacer uso de nuestro sistema de reserva.

```
// Funcion para la salida del programa.  
void salirPrograma() {  
    std::cout << "\nGracias por usar el sistema, vuelva pronto.\n";  
}  
  
// 'main' coordina las llamadas a las funciones, sin logica de negocio propia.  
int main() {  
    int opcion;  
    do {  
        std::cout << "\n--- Menu Principal - Gestion Turistica Galapagos ---\n";  
        std::cout << "1. Calcular costo de un tour\n";  
        std::cout << "2. Gestionar reserva de actividad\n";  
        std::cout << "3. Analizar datos turisticos\n";  
        std::cout << "4. Mostrar historial de reservas\n";  
        std::cout << "5. Salir\n";  
        opcion = obtenerEnteroValido("Seleccione una opcion (1-5): ", 1, 5);  
  
        switch (opcion) {  
            case 1:  
                calcularCostoTour();  
                break;  
            case 2:  
                gestionarReservaActividad();  
                break;  
            case 3:  
                analizarDatosTuristicos();  
                break;  
            case 4:  
                mostrarHistorialReservas();  
                break;  
            case 5:  
                salirPrograma();  
                break;  
        }  
    } while (opcion != 5);  
  
    return 0;  
}
```

Figura [15]: Código opción 5

- 7) Para terminar, realizaremos diagramas de flujo de las cinco opciones.

3. Casos para prueba

Para comprobar el correcto funcionamiento del sistema, seguimos algunos casos solicitados por nuestro docente.

Opción 1: Costo:

Isala=1

Personas=6

Tour=2

Temporada=2

Caso 1:

```
--- Menu Principal - Gestion Turistica Galapagos ---
1. Calcular costo de un tour
2. Gestionar reserva de actividad
3. Analizar datos turisticos
4. Mostrar historial de reservas
5. Salir
Seleccione una opcion (1-5): 1

--- Calcular Costo de Tour ---
Ingrese la isla (1-San Cristobal, 2-Santa Cruz, 3-Isabela): 1
Ingrese el numero de personas (1-100): 6
Ingrese el tipo de tour (1-Economico, 2-Premium): 2
Ingrese la temporada (1-Alta, 2-Baja): 2
El costo del tour para 6 personas en Isla San Cristobal es de $648.00.
```

Figura [16]: Caso 1

Opción 2: Reserva:

Snorkel en San Cristobal, día 1, 5, personas; buceo en Santa Cruz

Caso 2:

```
--- Menu Principal - Gestion Turistica Galapagos ---
1. Calcular costo de un tour
2. Gestionar reserva de actividad
3. Analizar datos turisticos
4. Mostrar historial de reservas
5. Salir
Seleccione una opcion (1-5): 2

--- Gestionar Reserva de Actividad ---
Ingrese la isla (1-San Cristobal, 2-Santa Cruz, 3-Isabela): 1
Ingrese la actividad (1-Snorkel, 2-Caminata, 3-Buceo, 4-Observacion de aves): 1
Ingrese el dia (1-7): 1
Ingrese el numero de personas (1-50): 5
Reserva exitosa para actividad Snorkel en Isla San Cristobal para el dia 1 (domingo) para 5 personas.
```

Figura [17]: Caso 2 (primera parte)

```
--- Menu Principal - Gestion Turistica Galapagos ---
1. Calcular costo de un tour
2. Gestionar reserva de actividad
3. Analizar datos turisticos
4. Mostrar historial de reservas
5. Salir
Seleccione una opcion (1-5): 2

--- Gestionar Reserva de Actividad ---
Ingrese la isla (1-San Cristobal, 2-Santa Cruz, 3-Isabela): 2
Ingrese la actividad (1-Snorkel, 2-Caminata, 3-Buceo, 4-Observacion de aves): 3
Ingrese el dia (1-7): 1
Ingrese el numero de personas (1-50): 5
Error: La actividad Buceo no esta permitida en la Isla Santa Cruz.
```

Figura [18]: Caso 2 (segunda parte)

Opción 3: Análisis:

Visitantes [10, 20, 15, 30, 25, 5, 10] para 3 Islas.

Caso 3:

```
1. Calcular costo de un tour
2. Gestionar reserva de actividad
3. Analizar datos turisticos
4. Mostrar historial de reservas
5. Salir
Seleccione una opcion (1-5): 3

--- Analizar Datos Turisticos ---
Ingrese el numero de visitantes para Isla San Cristobal (7 dias):
Dia 1 (0-1000): 10
Dia 2 (0-1000): 20
Dia 3 (0-1000): 15
Dia 4 (0-1000): 30
Dia 5 (0-1000): 25
Dia 6 (0-1000): 5
Dia 7 (0-1000): 10
Ingrese el numero de visitantes para Isla Santa Cruz (7 dias):
Dia 1 (0-1000): 10
Dia 2 (0-1000): 20
Dia 3 (0-1000): 15
Dia 4 (0-1000): 30
Dia 5 (0-1000): 25
Dia 6 (0-1000): 5
Dia 7 (0-1000): 10
Ingrese el numero de visitantes para Isla Isabela (7 dias):
Dia 1 (0-1000): 10
Dia 2 (0-1000): 20
Dia 3 (0-1000): 15
Dia 4 (0-1000): 30
Dia 5 (0-1000): 25
Dia 6 (0-1000): 5
Dia 7 (0-1000): 10
Estadisticas de Visitantes por Isla:
Isla          Promedio      Maximo      Minimo
-----  
San Cristobal    16.43        30            5
Santa Cruz       16.43        30            5
Isabela          16.43        30            5
-----  
Recomendacion: Visite la Isla San Cristobal (es la que, en promedio, tiene menor afluencia).
```

Figura [19]: Caso 3

Opción 4: Historial:

Mostrar 2 veces

Caso 4:

```
--- Menu Principal - Gestion Turistica Galapagos ---
1. Calcular costo de un tour
2. Gestionar reserva de actividad
3. Analizar datos turisticos
4. Mostrar historial de reservas
5. Salir
Seleccione una opcion (1-5): 4

--- Historial de Reservas ---
Reserva 1: Actividad de Snorkel en Isla San Cristobal para el dia 1 (domingo) para 5 personas.
```

Figura [20]: Caso 4

Referencias

Cecil, R. (2018). Código Limpio Manual de estilo para el desarrollo ágil de software [Archivo PDF].

<https://elhacker.info/manuales/Lenguajes%20de%20Programacion/Codigo%20limpio%20-%20Robert%20Cecil%20Martin.pdf.com>

Alberto Guzman. (2021, 7 diciembre). *Pseudocódigos y diagramas de flujo en PSeint* [Vídeo]. YouTube. <https://www.youtube.com/watch?v=KVkQxbg778c.com>