U.PORTO
FEUP FACULDADE DE ENGENHARIA
UNIVERSIDADE DO PORTO

PROJETO 2 - VIAGENS AÉREAS

ALGORITMOS E ESTRUTURAS DE DADOS

1º SEMESTRE - 2º ANO

Turma 12 - Grupo 8:
Luana Lima, 202206845
Lucas Greco, 202208296
Luís Cordeiro, 202205425

Índice

Visão geral

Este projeto tem como objetivo a criação de um programa em C++, capaz de gerir os voos para uma rede de viagens de companhias aéreas em todo o mundo, fornecendo assistência a utilizadores que desejem explorar e planear viagens.

Estruturas de Dados - Grafos

CLASSE GRAPH

Modelação de um grafo de aeroportos e voos, com funcionalidades como busca de aeroportos, adição de novos aeroportos e voos, estatísticas sobre o número de voos, análise de conectividade e identificação de aeroportos essenciais. Utiliza estruturas eficientes, como vetores e mapas, suportando operações como encontrar conexões rápidas entre aeroportos, determinar o diâmetro máximo do grafo e listar os principais aeroportos.

Classes

AIRLINE

Armazena informações sobre as companhias aéreas, incluindo o seu código, nome, indicativo e país de origem.

AIRPORT

Armazena informações sobre os aeroportos, incluindo o seu código, nome, cidade e país de origem, latitude e longitude, fornecendo métodos para adicionar voos, marcar o aeroporto como visitado, verificar o número de voos e se foi processado.

FLIGHT

Mantém informações sobre voos, como a companhia aérea e o código do aeroporto de destino.

GRAPH

Nesta classe, são manipulados dados referentes a voos e aeroportos, com várias funcionalidades que permitem a sua análise.

SYSTEM

É a classe central do programa, pois aqui, todas as outras classes interagem. É responsável por ler arquivos, bem como armazenar listas de aeroportos e companhias aéreas.

MENU

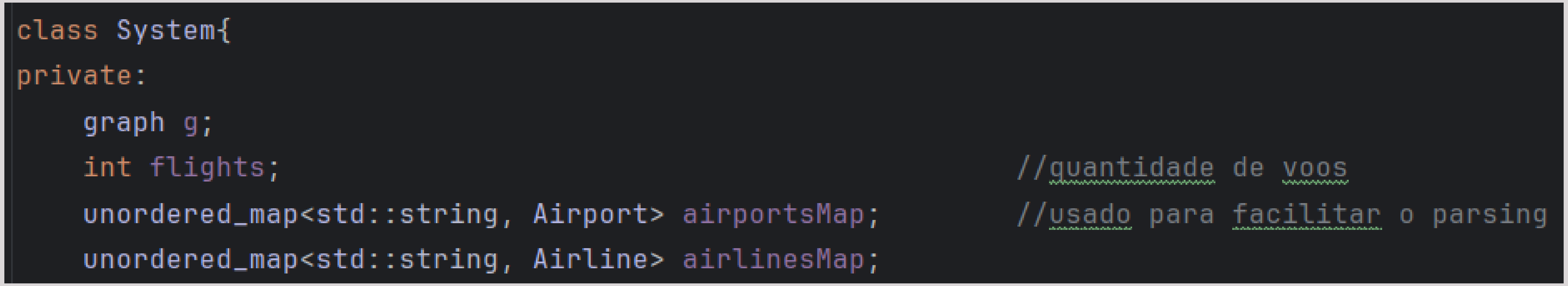
Esta classe permite que o usuário interaja com as funcionalidades do programa e exibe as saídas correspondentes.

Funcionalidades

- Apresentar as melhores opções de voos
- Estatísticas sobre voos sob diferentes condições
- Procurar as viagens com o maior número de paragens
- Identificar os aeroportos com maior número de voos
- Procurar os melhores voos com base em filtros
- Identificar os aeroportos essenciais à circulação da rede

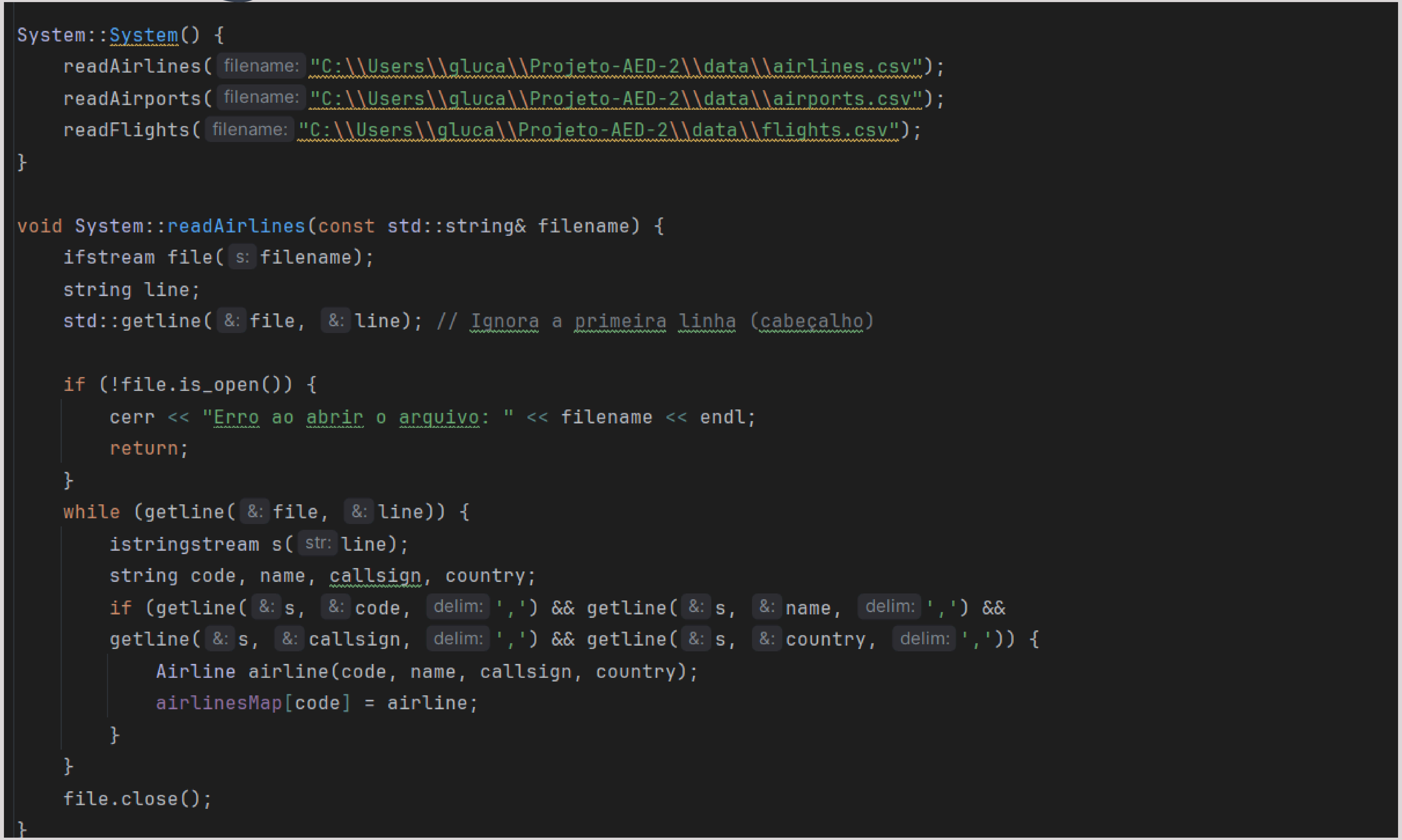
Exemplos de Código

System

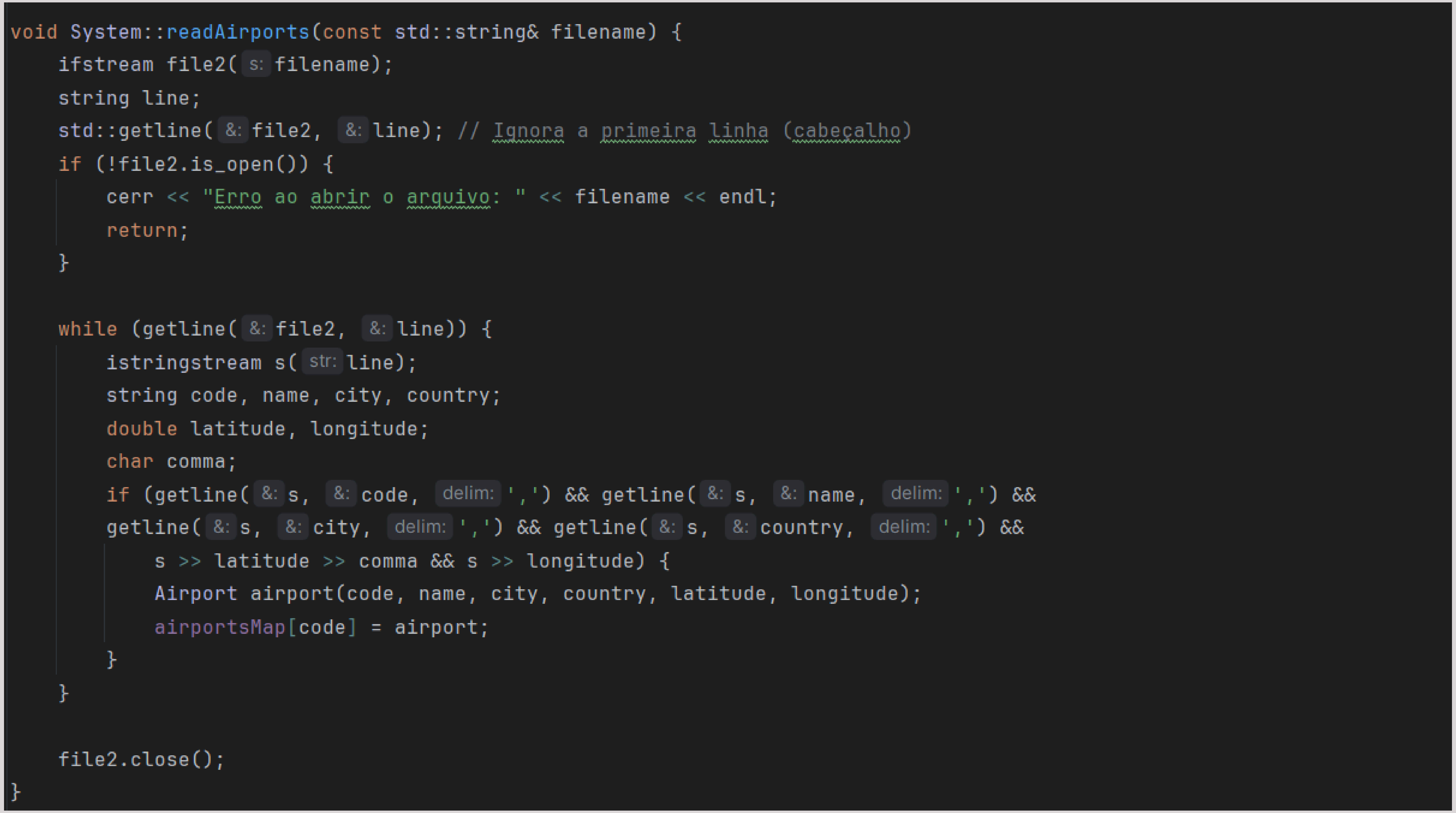


```
class System{
private:
    graph g;
    int flights;                                            //quantidade de voos
    unordered_map<std::string, Airport> airportsMap;        //usado para facilitar o parsing
    unordered_map<std::string, Airline> airlinesMap;
```

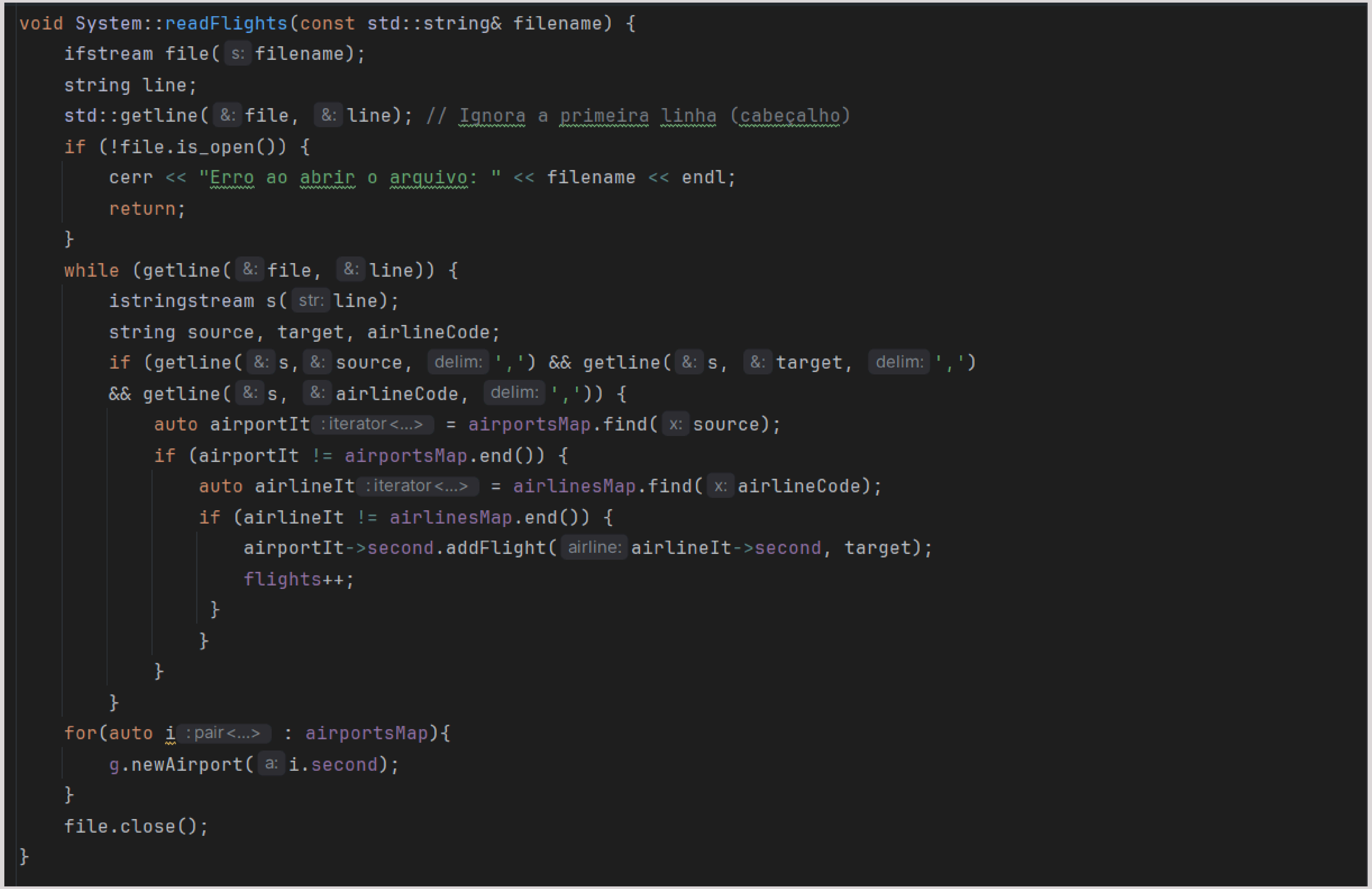
Parsing



```cpp
System::System() {
    readAirlines("C:\\Users\\gluca\\Projeto-AED-2\\data\\airlines.csv");
    readAirports("C:\\Users\\gluca\\Projeto-AED-2\\data\\airports.csv");
    readFlights("C:\\Users\\gluca\\Projeto-AED-2\\data\\flights.csv");
}

void System::readAirlines(const std::string& filename) {
    ifstream file(filename);
    string line;
    std::getline(file, line); // Ignora a primeira linha (cabeçalho)

    if (!file.is_open()) {
        cerr << "Erro ao abrir o arquivo: " << filename << endl;
        return;
    }
    while (getline(file, line)) {
        istringstream s(line);
        string code, name, callsign, country;
        if (getline(s, code, ',') && getline(s, name, ',') &&
        getline(s, callsign, ',') && getline(s, country, ',')) {
            Airline airline(code, name, callsign, country);
            airlinesMap[code] = airline;
        }
    }
    file.close();
}
```

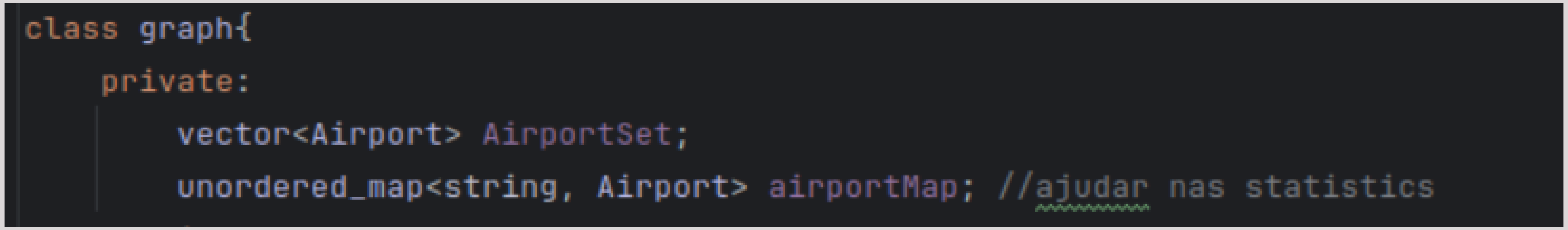


```
void System::readAirports(const std::string& filename) {
    ifstream file2( s: filename);
    string line;
    std::getline( &: file2,  &: line); // Ignora a primeira linha (cabeçalho)
    if (!file2.is_open()) {
        cerr << "Erro ao abrir o arquivo: " << filename << endl;
        return;
    }

    while (getline( &: file2,  &: line)) {
        istringstream s( str: line);
        string code, name, city, country;
        double latitude, longitude;
        char comma;
        if (getline( &: s,  &: code,  delim: ',') && getline( &: s,  &: name,  delim: ',') &&
        getline( &: s,  &: city,  delim: ',') && getline( &: s,  &: country,  delim: ',') &&
            s >> latitude >> comma && s >> longitude) {
            Airport airport(code, name, city, country, latitude, longitude);
            airportsMap[code] = airport;
        }
    }

    file2.close();
}
```

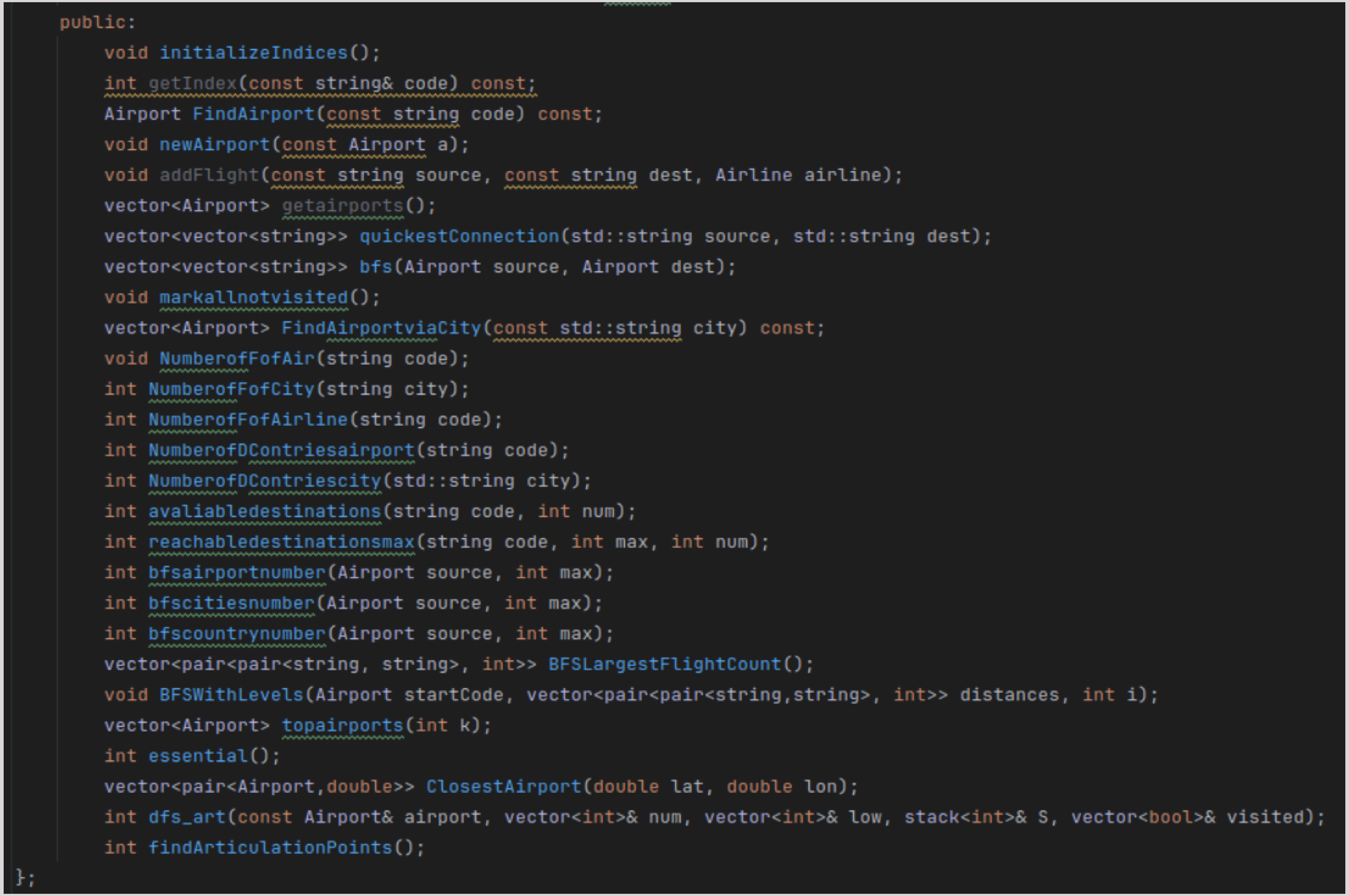


```
void System::readFlights(const std::string& filename) {
    ifstream file( s: filename);
    string line;
    std::getline( &: file,  &: line); // Ignora a primeira linha (cabeçalho)
    if (!file.is_open()) {
        cerr << "Erro ao abrir o arquivo: " << filename << endl;
        return;
    }
    while (getline( &: file,  &: line)) {
        istringstream s( str: line);
        string source, target, airlineCode;
        if (getline( &: s, &: source,  delim: ',') && getline( &: s,  &: target,  delim: ',')
        && getline( &: s,  &: airlineCode,  delim: ',')) {
            auto airportIt :iterator<...> = airportsMap.find( x: source);
            if (airportIt != airportsMap.end()) {
                auto airlineIt :iterator<...> = airlinesMap.find( x: airlineCode);
                if (airlineIt != airlinesMap.end()) {
                    airportIt->second.addFlight( airline: airlineIt->second, target);
                    flights++;
                }
            }
        }
    }
    for(auto i :pair<...> : airportsMap){
        g.newAirport( a: i.second);
    }
    file.close();
}
```

Classe Graph

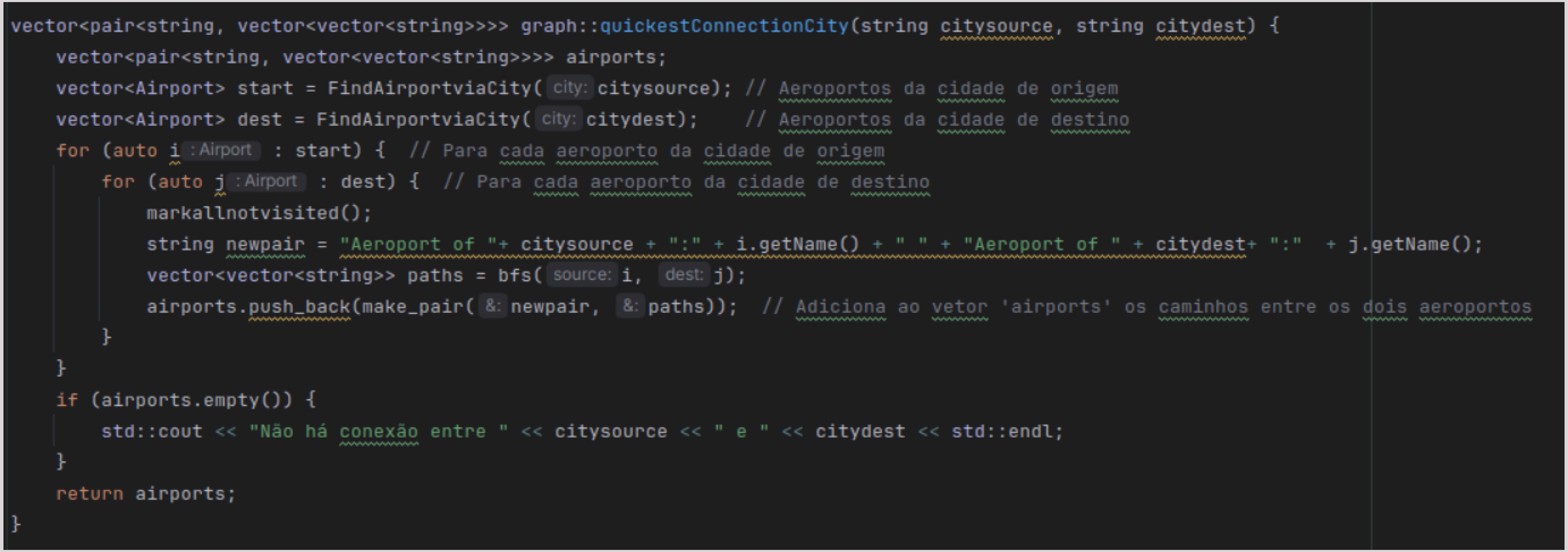


```
class graph{
    private:
        vector<Airport> AirportSet;
        unordered_map<string, Airport> airportMap; //ajudar nas statistics
```



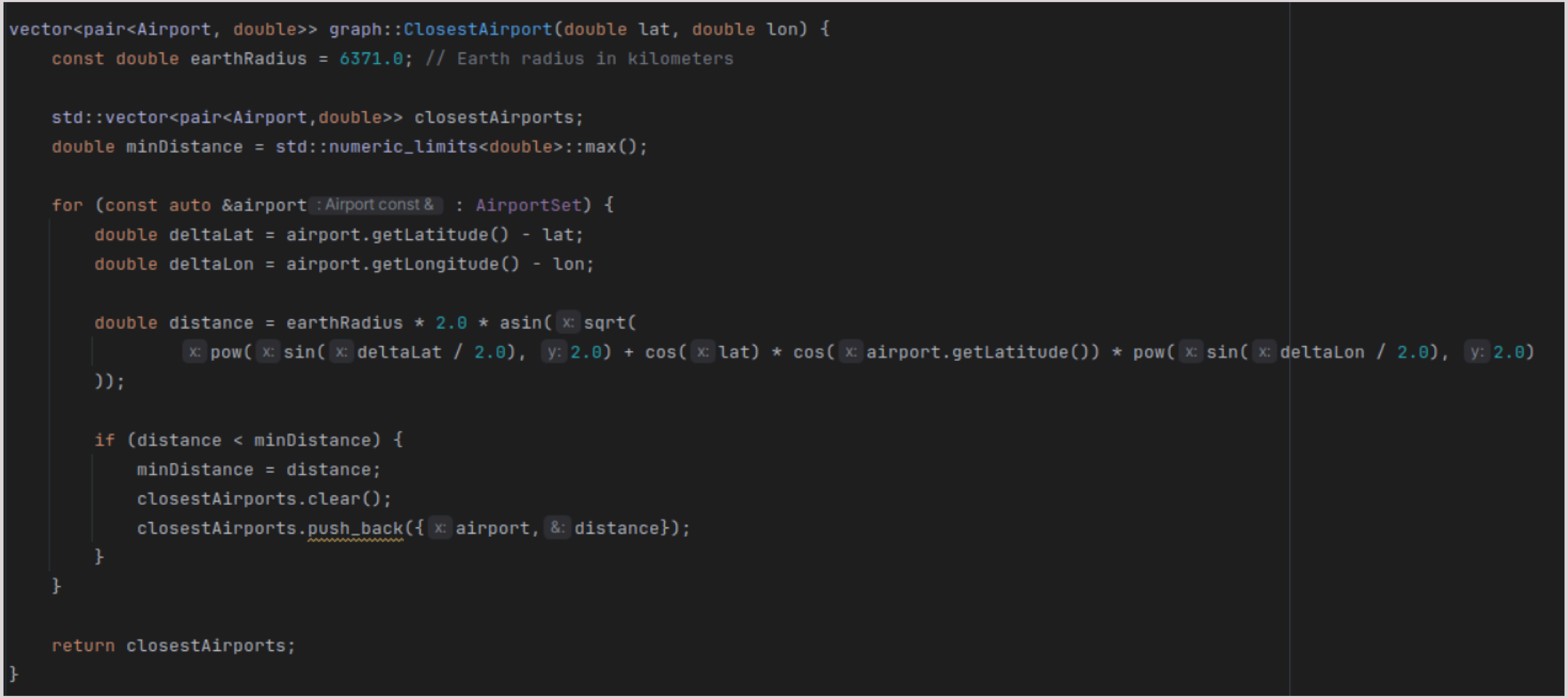
```
    public:
        void initializeIndices();
        int getIndex(const string& code) const;
        Airport FindAirport(const string code) const;
        void newAirport(const Airport a);
        void addFlight(const string source, const string dest, Airline airline);
        vector<Airport> getairports();
        vector<vector<string>> quickestConnection(std::string source, std::string dest);
        vector<vector<string>> bfs(Airport source, Airport dest);
        void markallnotvisited();
        vector<Airport> FindAirportviaCity(const std::string city) const;
        void NumberofFofAir(string code);
        int NumberofFofCity(string city);
        int NumberofFofAirline(string code);
        int NumberofDContriesairport(string code);
        int NumberofDContriescity(std::string city);
        int avaliabledestinations(string code, int num);
        int reachabledestinationsmax(string code, int max, int num);
        int bfsairportnumber(Airport source, int max);
        int bfscitiesnumber(Airport source, int max);
        int bfscountrynumber(Airport source, int max);
        vector<pair<pair<string, string>, int>> BFSLargestFlightCount();
        void BFSWithLevels(Airport startCode, vector<pair<pair<string,string>, int>> distances, int i);
        vector<Airport> topairports(int k);
        int essential();
        vector<pair<Airport,double>> ClosestAirport(double lat, double lon);
        int dfs_art(const Airport& airport, vector<int>& num, vector<int>& low, stack<int>& S, vector<bool>& visited);
        int findArticulationPoints();
};
```

O(V^2) - V é o número de aeroportos do AeroportSet



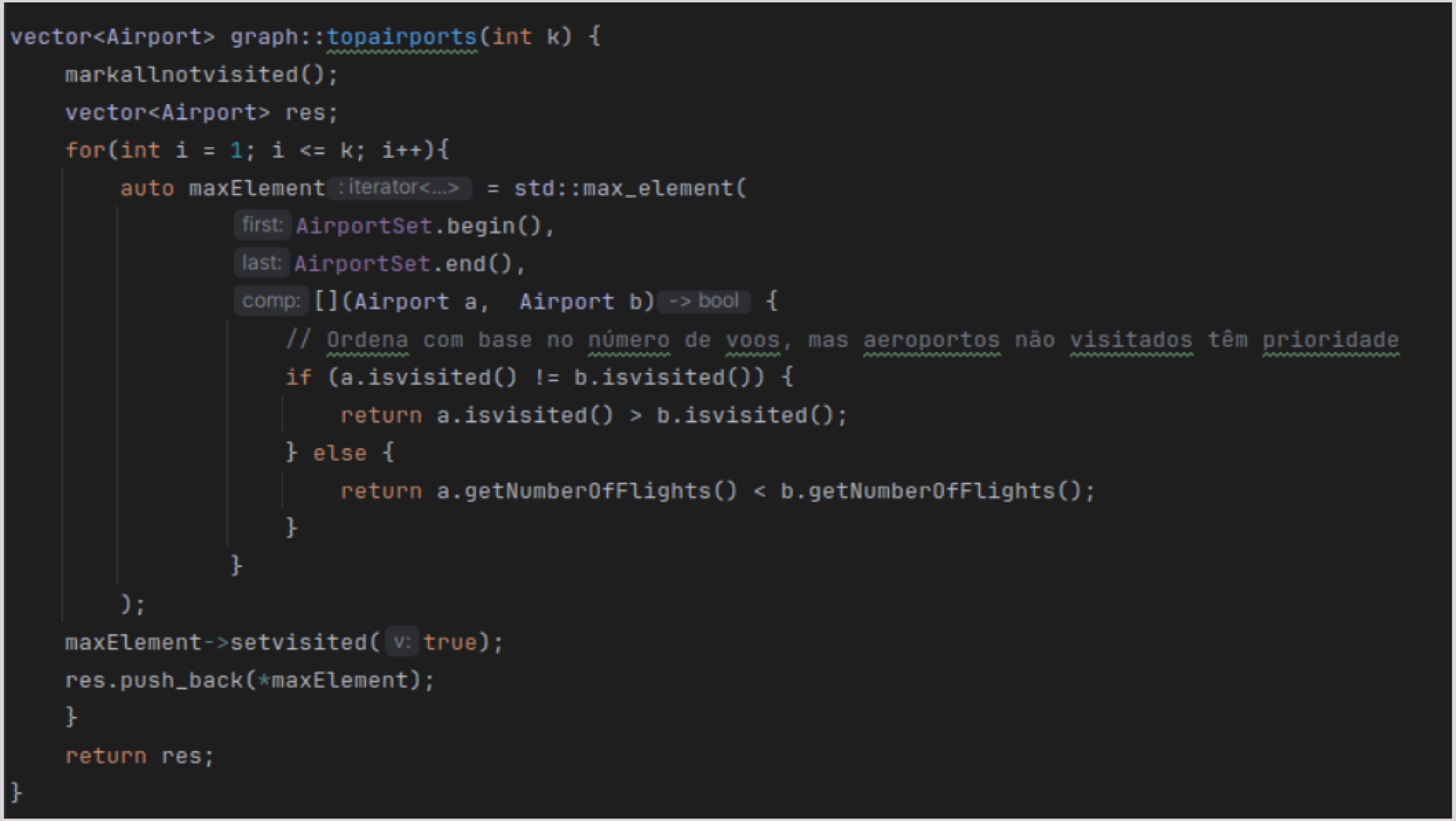
```
vector<pair<string, vector<vector<string>>>> graph::quickestConnectionCity(string citysource, string citydest) {
    vector<pair<string, vector<vector<string>>>> airports;
    vector<Airport> start = FindAirportviaCity( city: citysource); // Aeroportos da cidade de origem
    vector<Airport> dest = FindAirportviaCity( city: citydest);    // Aeroportos da cidade de destino
    for (auto i : Airport  : start) {  // Para cada aeroporto da cidade de origem
        for (auto j : Airport  : dest) {  // Para cada aeroporto da cidade de destino
            markallnotvisited();
            string newpair = "Aeroport of "+ citysource + ":" + i.getName() + " " + "Aeroport of " + citydest+ ":"  + j.getName();
            vector<vector<string>> paths = bfs( source: i,  dest: j);
            airports.push_back(make_pair( &: newpair,  &: paths));  // Adiciona ao vetor 'airports' os caminhos entre os dois aeroportos
        }
    }
    if (airports.empty()) {
        std::cout << "Não há conexão entre " << citysource << " e " << citydest << std::endl;
    }
    return airports;
}
```

O(V)



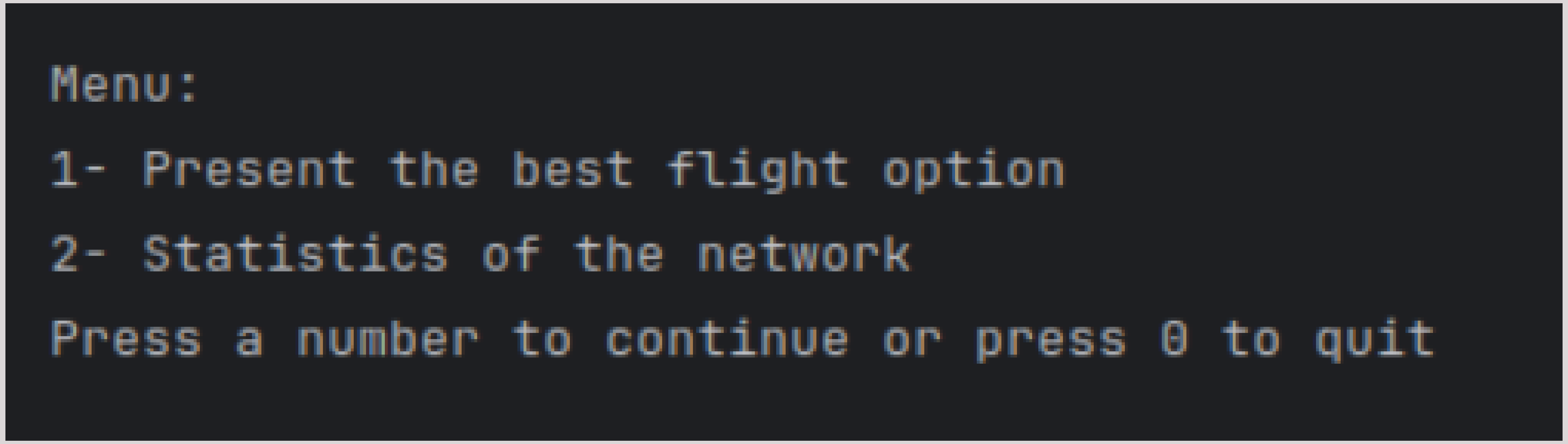
```
vector<pair<Airport, double>> graph::ClosestAirport(double lat, double lon) {
    const double earthRadius = 6371.0; // Earth radius in kilometers

    std::vector<pair<Airport,double>> closestAirports;
    double minDistance = std::numeric_limits<double>::max();

    for (const auto &airport : Airport const &  : AirportSet) {
        double deltaLat = airport.getLatitude() - lat;
        double deltaLon = airport.getLongitude() - lon;

        double distance = earthRadius * 2.0 * asin( x: sqrt(
                x: pow( x: sin( x: deltaLat / 2.0),  y: 2.0) + cos( x: lat) * cos( x: airport.getLatitude()) * pow( x: sin( x: deltaLon / 2.0),  y: 2.0)
        ));

        if (distance < minDistance) {
            minDistance = distance;
            closestAirports.clear();
            closestAirports.push_back({ x: airport, &: distance});
        }
    }

    return closestAirports;
}
```

O(k * (V + log(V)))

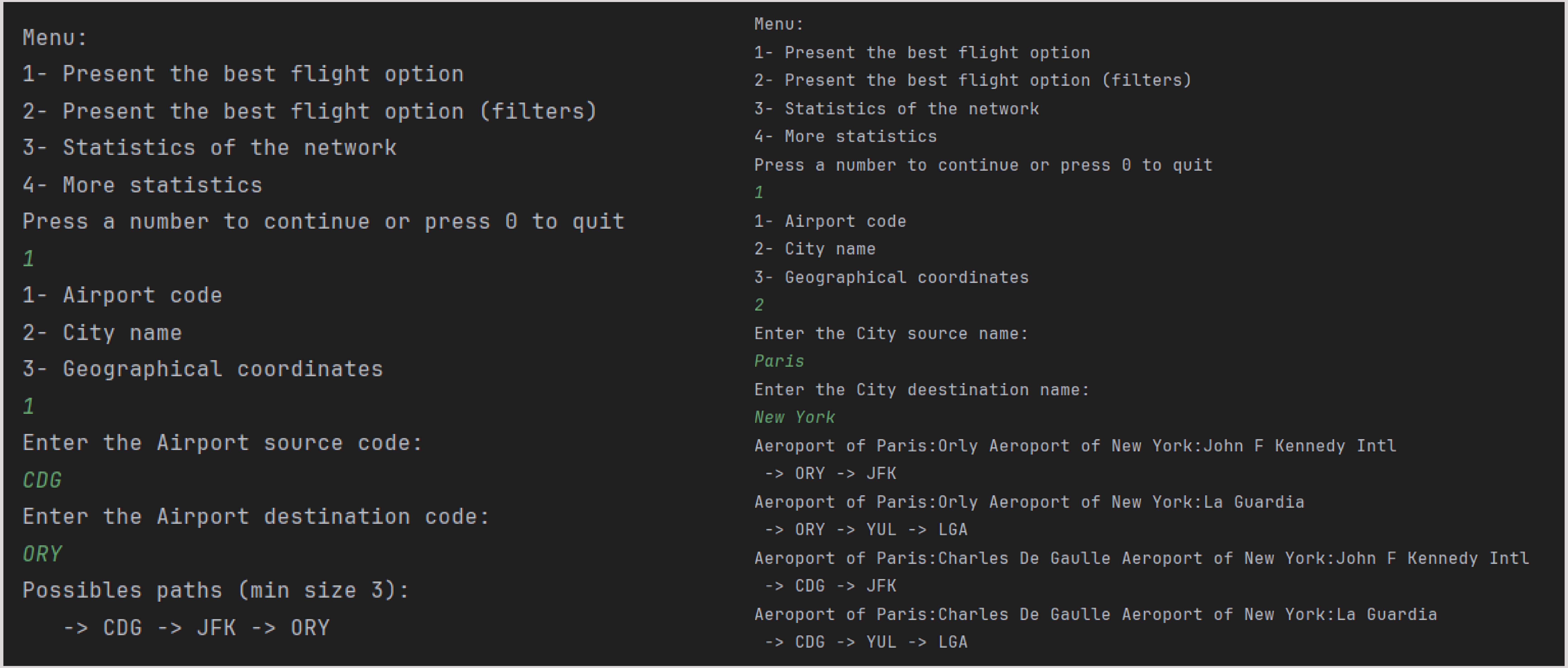


```
vector<Airport> graph::topairports(int k) {
    markallnotvisited();
    vector<Airport> res;
    for(int i = 1; i <= k; i++){
        auto maxElement :iterator<...> = std::max_element(
                first: AirportSet.begin(),
                last: AirportSet.end(),
                comp: [](Airport a,  Airport b) -> bool  {
                    // Ordena com base no número de voos, mas aeroportos não visitados têm prioridade
                    if (a.isvisited() != b.isvisited()) {
                        return a.isvisited() > b.isvisited();
                    } else {
                        return a.getNumberOfFlights() < b.getNumberOfFlights();
                    }
                }
        );
    maxElement->setvisited( v: true);
    res.push_back(*maxElement);
    }
    return res;
}
```

Exemplos de Execução

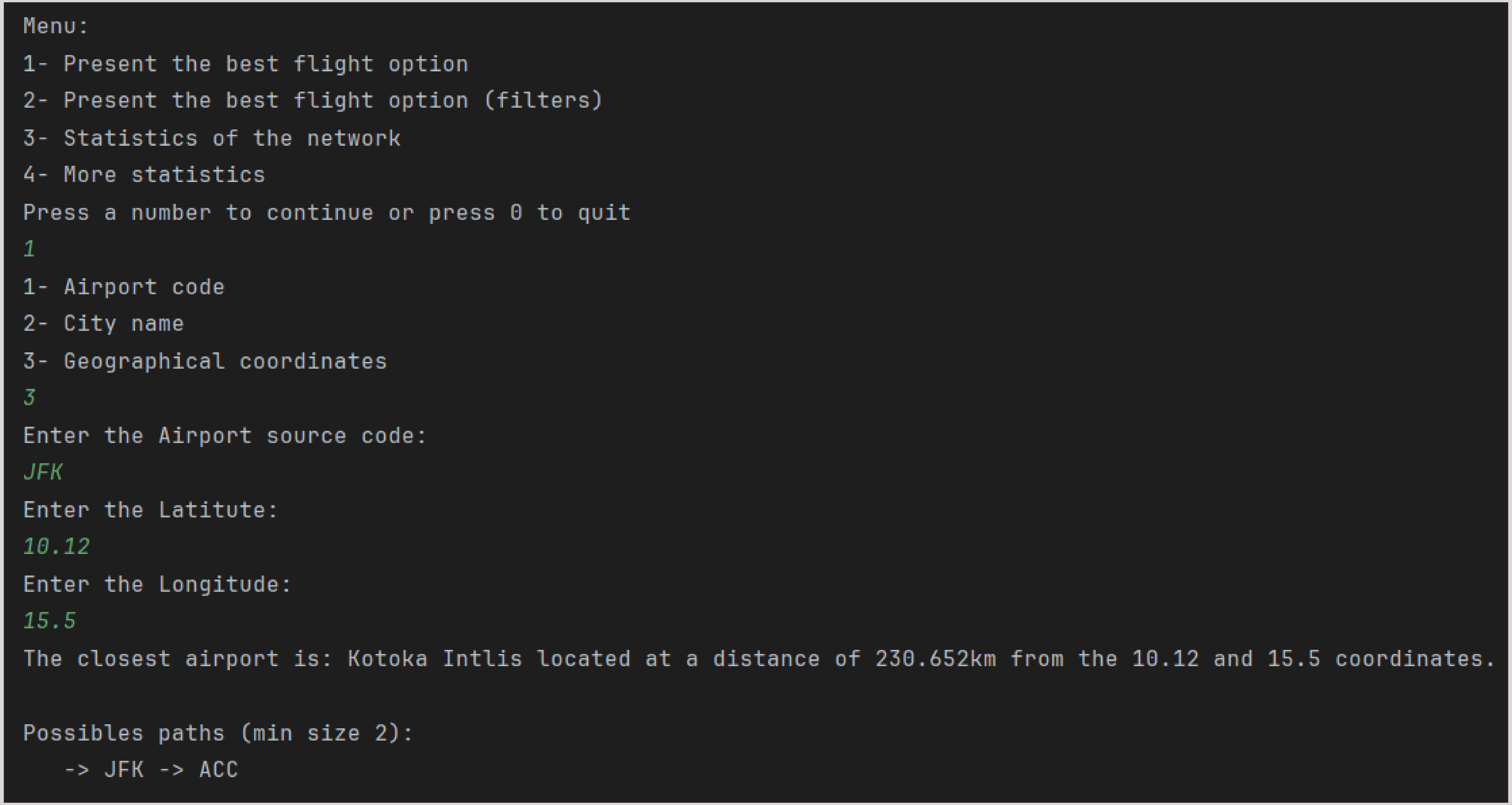


```
Menu:
1- Present the best flight option
2- Statistics of the network
Press a number to continue or press 0 to quit
```

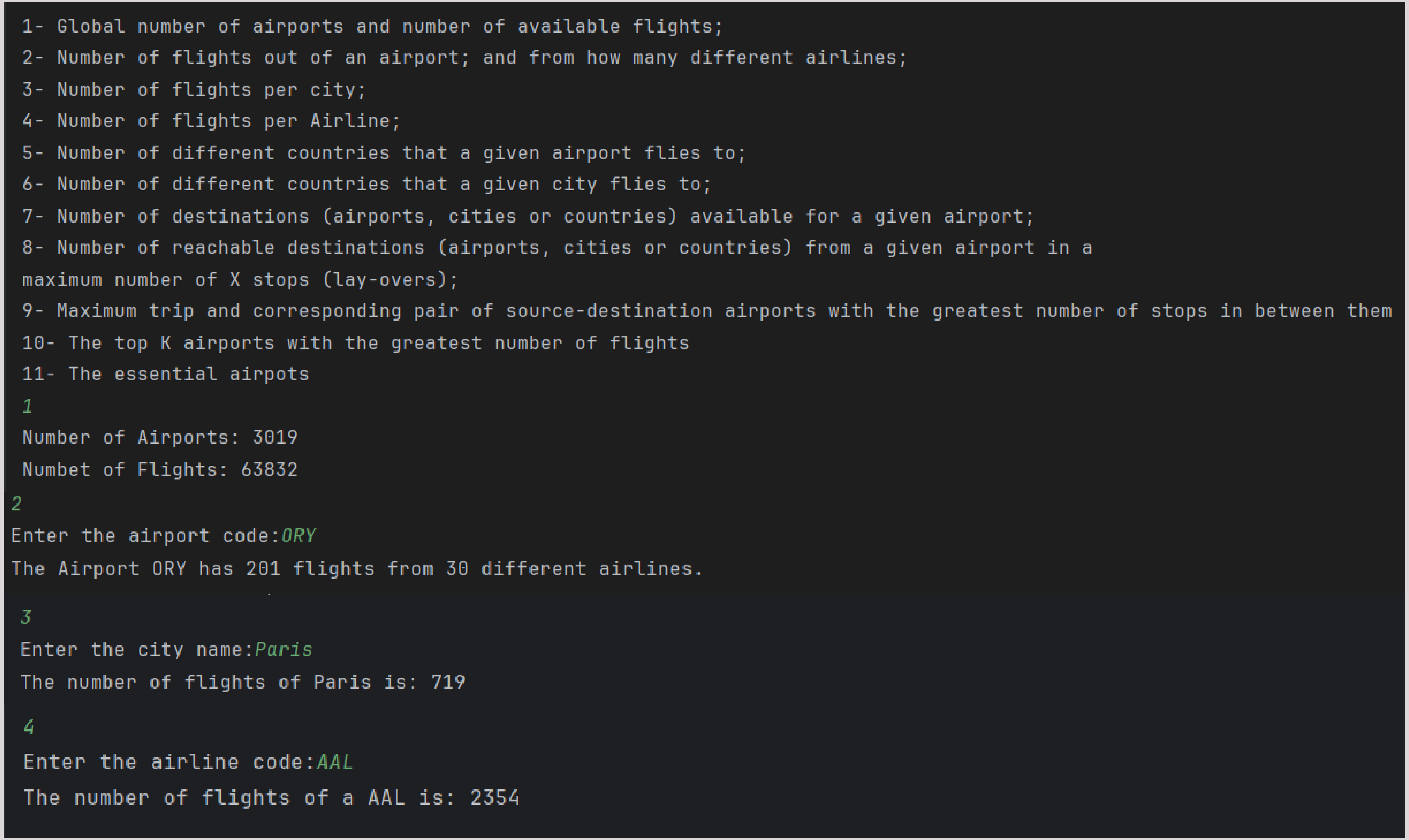


```
Menu:
1- Present the best flight option
2- Present the best flight option (filters)
3- Statistics of the network
4- More statistics
Press a number to continue or press 0 to quit
1
1- Airport code
2- City name
3- Geographical coordinates
1
Enter the Airport source code:
CDG
Enter the Airport destination code:
ORY
Possibles paths (min size 3):
   -> CDG -> JFK -> ORY
```

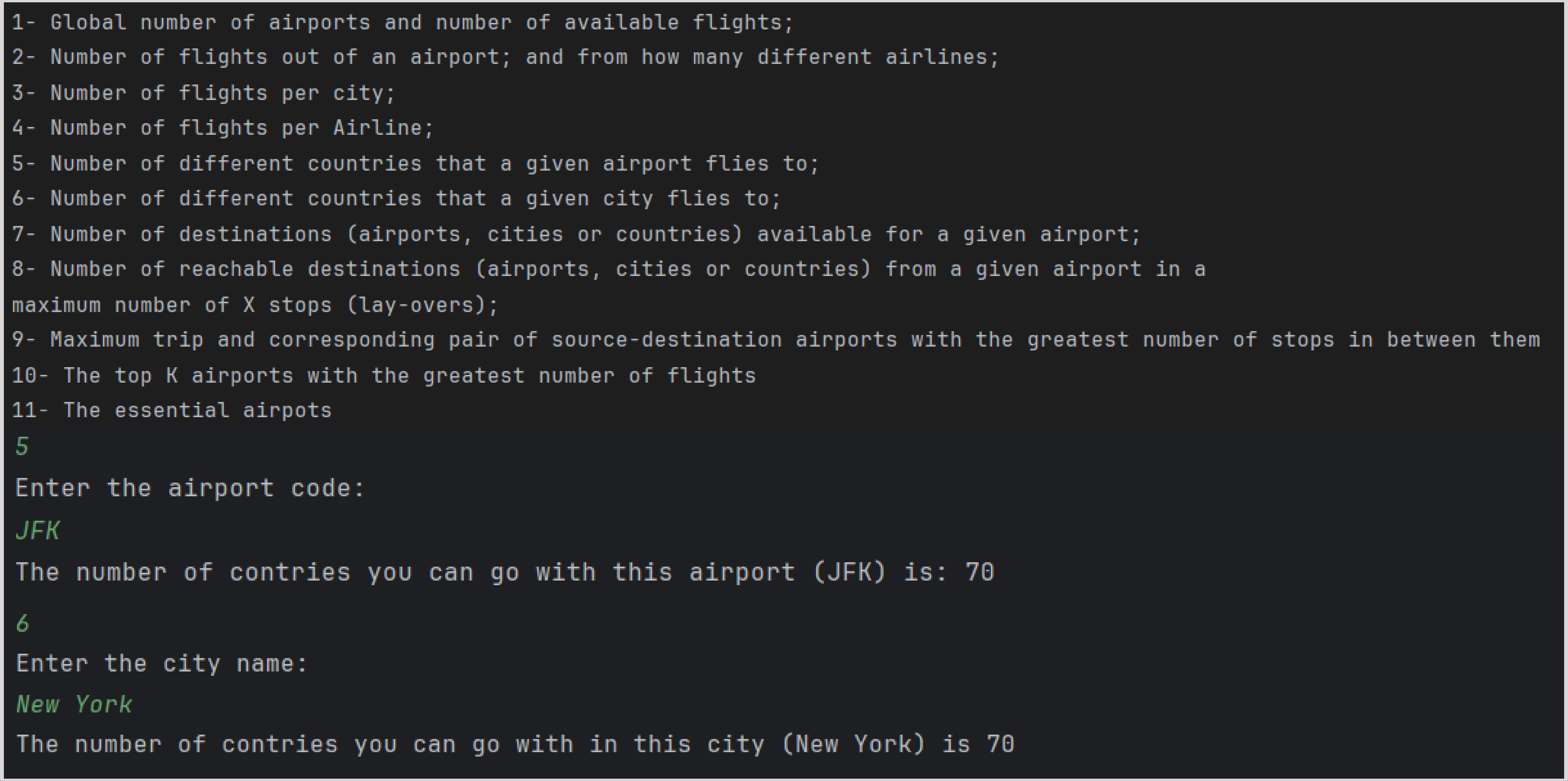
```
Menu:
1- Present the best flight option
2- Present the best flight option (filters)
3- Statistics of the network
4- More statistics
Press a number to continue or press 0 to quit
1
1- Airport code
2- City name
3- Geographical coordinates
2
Enter the City source name:
Paris
Enter the City deestination name:
New York
Aeroport of Paris:Orly Aeroport of New York:John F Kennedy Intl
 -> ORY -> JFK
Aeroport of Paris:Orly Aeroport of New York:La Guardia
 -> ORY -> YUL -> LGA
Aeroport of Paris:Charles De Gaulle Aeroport of New York:John F Kennedy Intl
 -> CDG -> JFK
Aeroport of Paris:Charles De Gaulle Aeroport of New York:La Guardia
 -> CDG -> YUL -> LGA
```



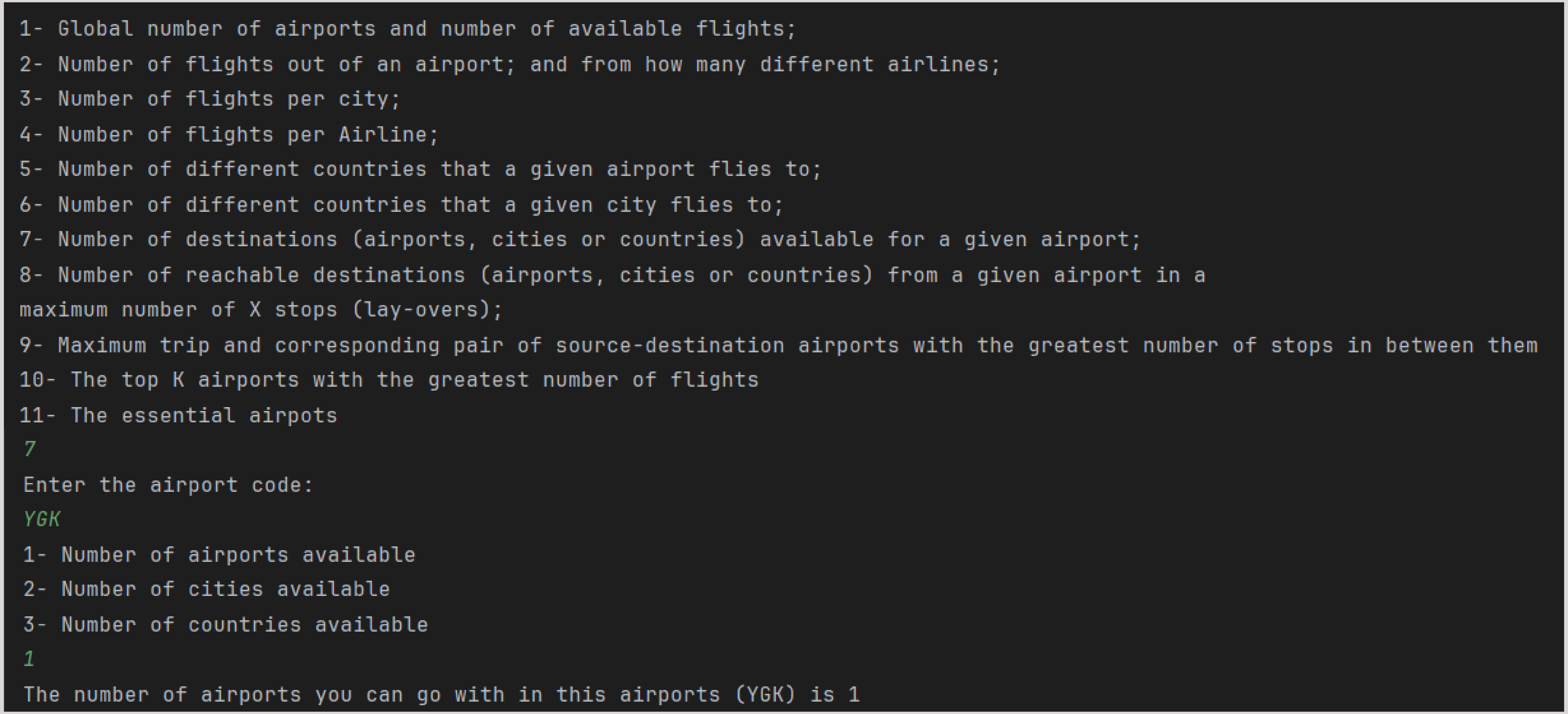
```
Menu:
1- Present the best flight option
2- Present the best flight option (filters)
3- Statistics of the network
4- More statistics
Press a number to continue or press 0 to quit
1
1- Airport code
2- City name
3- Geographical coordinates
3
Enter the Airport source code:
JFK
Enter the Latitute:
10.12
Enter the Longitude:
15.5
The closest airport is: Kotoka Intlis located at a distance of 230.652km from the 10.12 and 15.5 coordinates.

Possibles paths (min size 2):
   -> JFK -> ACC
```



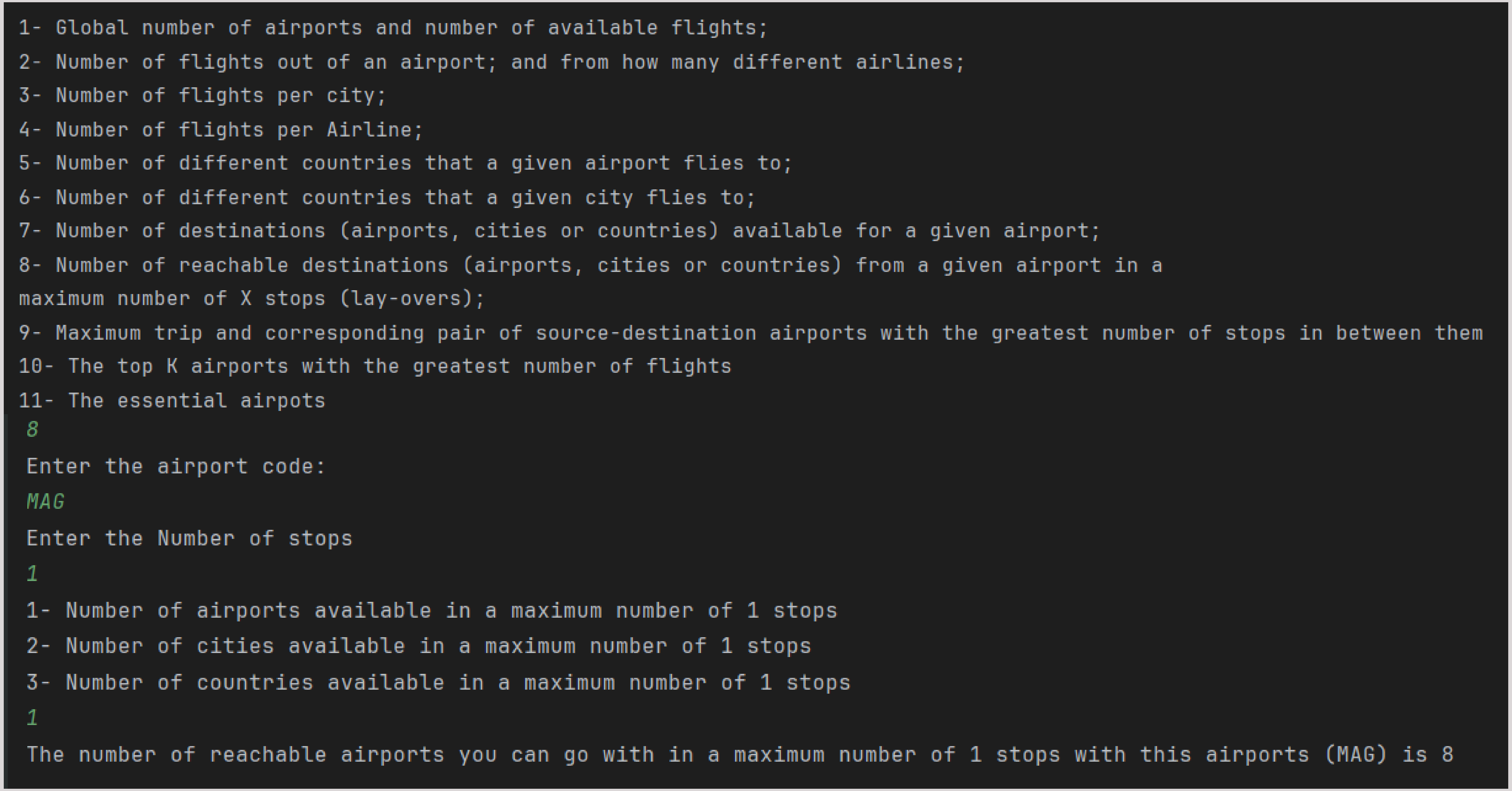
```
1- Global number of airports and number of available flights;
2- Number of flights out of an airport; and from how many different airlines;
3- Number of flights per city;
4- Number of flights per Airline;
5- Number of different countries that a given airport flies to;
6- Number of different countries that a given city flies to;
7- Number of destinations (airports, cities or countries) available for a given airport;
8- Number of reachable destinations (airports, cities or countries) from a given airport in a
maximum number of X stops (lay-overs);
9- Maximum trip and corresponding pair of source-destination airports with the greatest number of stops in between them
10- The top K airports with the greatest number of flights
11- The essential airpots
1
Number of Airports: 3019
Numbet of Flights: 63832
2
Enter the airport code:ORY
The Airport ORY has 201 flights from 30 different airlines.
3
Enter the city name:Paris
The number of flights of Paris is: 719
4
Enter the airline code:AAL
The number of flights of a AAL is: 2354
```



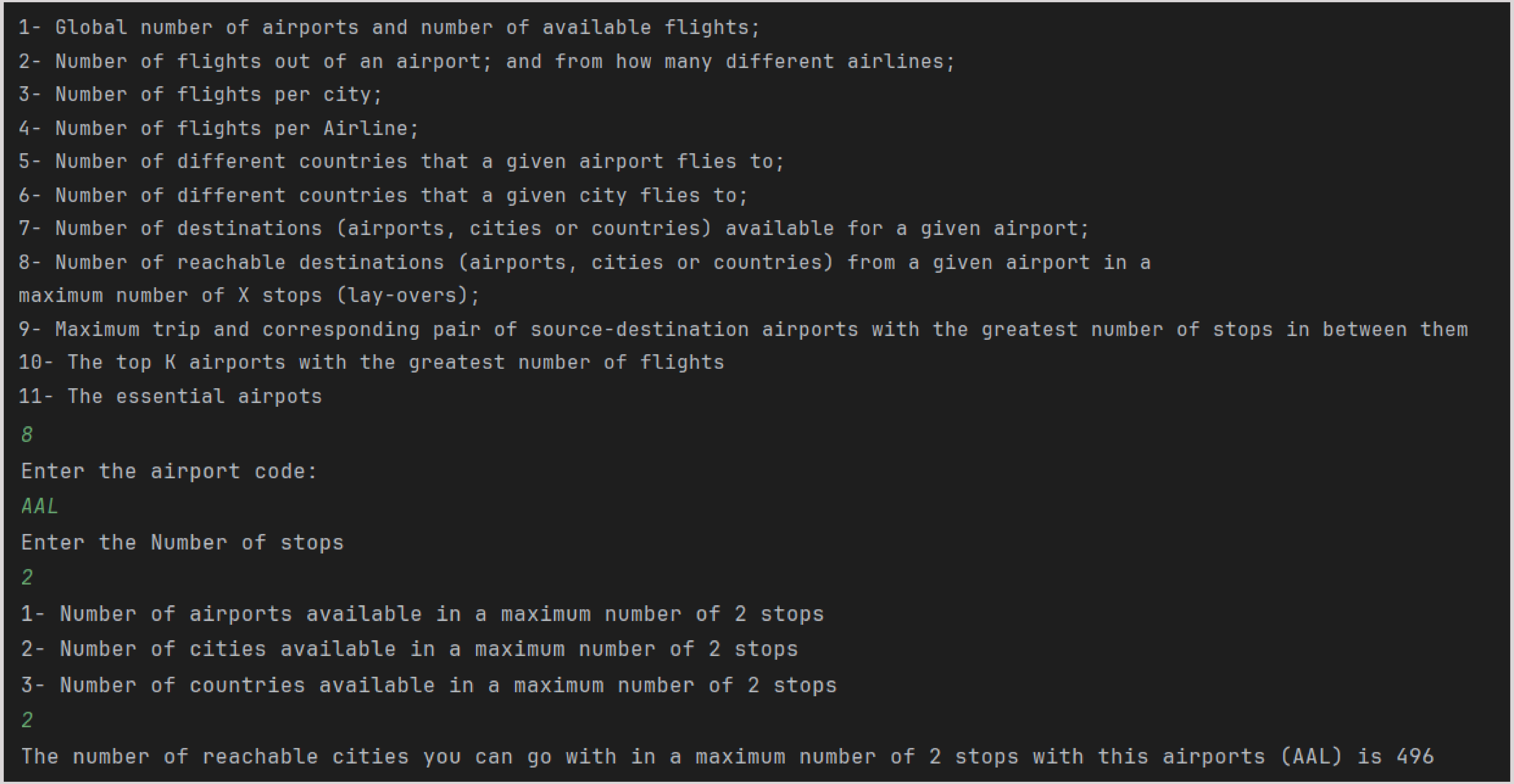
```
1- Global number of airports and number of available flights;
2- Number of flights out of an airport; and from how many different airlines;
3- Number of flights per city;
4- Number of flights per Airline;
5- Number of different countries that a given airport flies to;
6- Number of different countries that a given city flies to;
7- Number of destinations (airports, cities or countries) available for a given airport;
8- Number of reachable destinations (airports, cities or countries) from a given airport in a
maximum number of X stops (lay-overs);
9- Maximum trip and corresponding pair of source-destination airports with the greatest number of stops in between them
10- The top K airports with the greatest number of flights
11- The essential airpots
5
Enter the airport code:
JFK
The number of contries you can go with this airport (JFK) is: 70

6
Enter the city name:
New York
The number of contries you can go with in this city (New York) is 70
```



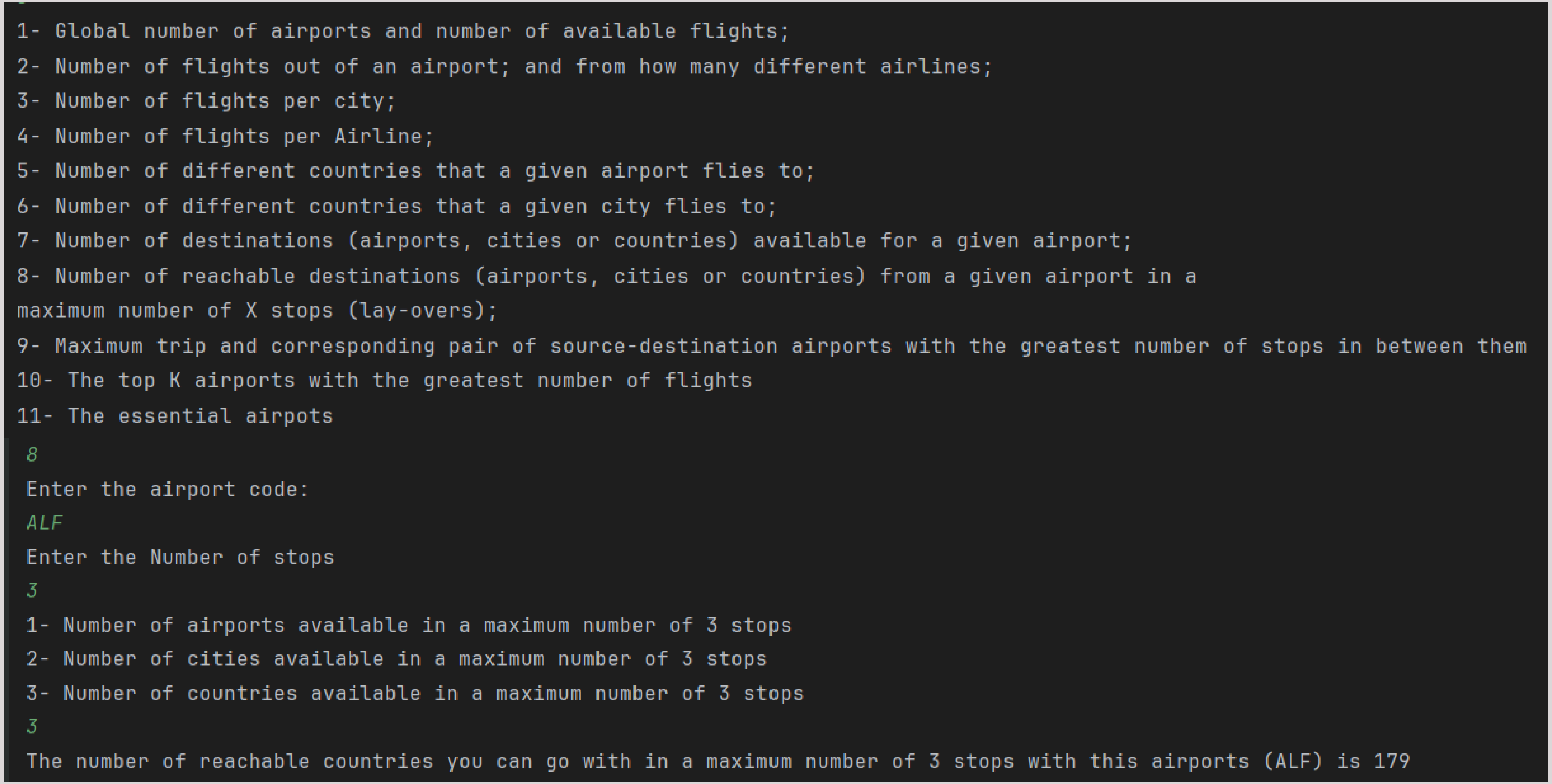
```
1- Global number of airports and number of available flights;
2- Number of flights out of an airport; and from how many different airlines;
3- Number of flights per city;
4- Number of flights per Airline;
5- Number of different countries that a given airport flies to;
6- Number of different countries that a given city flies to;
7- Number of destinations (airports, cities or countries) available for a given airport;
8- Number of reachable destinations (airports, cities or countries) from a given airport in a
maximum number of X stops (lay-overs);
9- Maximum trip and corresponding pair of source-destination airports with the greatest number of stops in between them
10- The top K airports with the greatest number of flights
11- The essential airpots
 7
 Enter the airport code:
 YGK
 1- Number of airports available
 2- Number of cities available
 3- Number of countries available
 1
 The number of airports you can go with in this airports (YGK) is 1
```



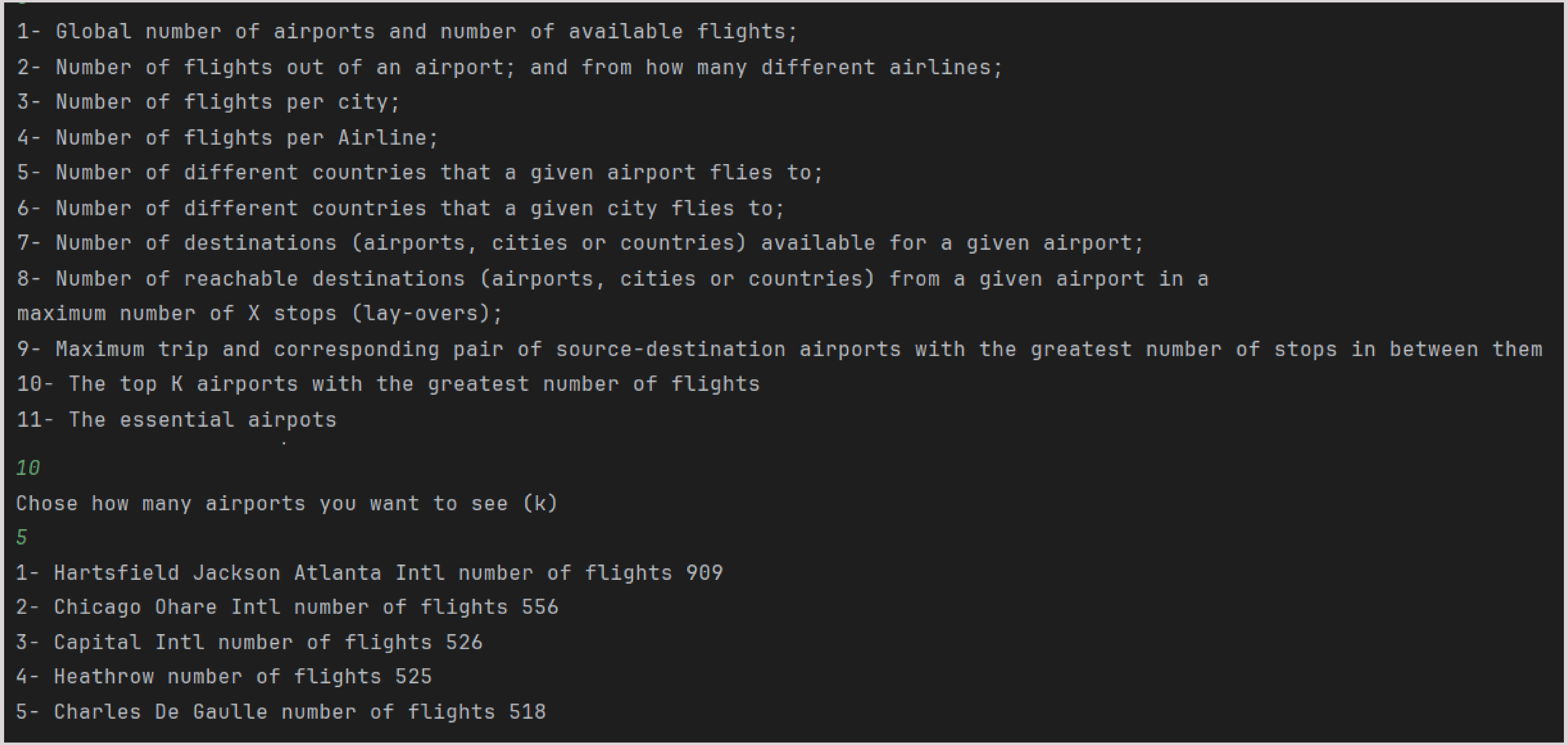
```
1- Global number of airports and number of available flights;
2- Number of flights out of an airport; and from how many different airlines;
3- Number of flights per city;
4- Number of flights per Airline;
5- Number of different countries that a given airport flies to;
6- Number of different countries that a given city flies to;
7- Number of destinations (airports, cities or countries) available for a given airport;
8- Number of reachable destinations (airports, cities or countries) from a given airport in a
maximum number of X stops (lay-overs);
9- Maximum trip and corresponding pair of source-destination airports with the greatest number of stops in between them
10- The top K airports with the greatest number of flights
11- The essential airpots
 8
 Enter the airport code:
 MAG
 Enter the Number of stops
 1
 1- Number of airports available in a maximum number of 1 stops
 2- Number of cities available in a maximum number of 1 stops
 3- Number of countries available in a maximum number of 1 stops
 1
 The number of reachable airports you can go with in a maximum number of 1 stops with this airports (MAG) is 8
```



```
1- Global number of airports and number of available flights;
2- Number of flights out of an airport; and from how many different airlines;
3- Number of flights per city;
4- Number of flights per Airline;
5- Number of different countries that a given airport flies to;
6- Number of different countries that a given city flies to;
7- Number of destinations (airports, cities or countries) available for a given airport;
8- Number of reachable destinations (airports, cities or countries) from a given airport in a
maximum number of X stops (lay-overs);
9- Maximum trip and corresponding pair of source-destination airports with the greatest number of stops in between them
10- The top K airports with the greatest number of flights
11- The essential airpots
8
Enter the airport code:
AAL
Enter the Number of stops
2
1- Number of airports available in a maximum number of 2 stops
2- Number of cities available in a maximum number of 2 stops
3- Number of countries available in a maximum number of 2 stops
2
The number of reachable cities you can go with in a maximum number of 2 stops with this airports (AAL) is 496
```



```
1- Global number of airports and number of available flights;
2- Number of flights out of an airport; and from how many different airlines;
3- Number of flights per city;
4- Number of flights per Airline;
5- Number of different countries that a given airport flies to;
6- Number of different countries that a given city flies to;
7- Number of destinations (airports, cities or countries) available for a given airport;
8- Number of reachable destinations (airports, cities or countries) from a given airport in a
maximum number of X stops (lay-overs);
9- Maximum trip and corresponding pair of source-destination airports with the greatest number of stops in between them
10- The top K airports with the greatest number of flights
11- The essential airpots
8
Enter the airport code:
ALF
Enter the Number of stops
3
1- Number of airports available in a maximum number of 3 stops
2- Number of cities available in a maximum number of 3 stops
3- Number of countries available in a maximum number of 3 stops
3
The number of reachable countries you can go with in a maximum number of 3 stops with this airports (ALF) is 179
```



```
1- Global number of airports and number of available flights;
2- Number of flights out of an airport; and from how many different airlines;
3- Number of flights per city;
4- Number of flights per Airline;
5- Number of different countries that a given airport flies to;
6- Number of different countries that a given city flies to;
7- Number of destinations (airports, cities or countries) available for a given airport;
8- Number of reachable destinations (airports, cities or countries) from a given airport in a
maximum number of X stops (lay-overs);
9- Maximum trip and corresponding pair of source-destination airports with the greatest number of stops in between them
10- The top K airports with the greatest number of flights
11- The essential airpots

10
Chose how many airports you want to see (k)
5
1- Hartsfield Jackson Atlanta Intl number of flights 909
2- Chicago Ohare Intl number of flights 556
3- Capital Intl number of flights 526
4- Heathrow number of flights 525
5- Charles De Gaulle number of flights 518
```

Principais Dificuldades

IMPLEMENTAR FILTROS

Ao tentarmos implementar filtros na procura da melhor opção de voo encontramos bastantes dificuldades

ENCONTRAR OS AEROPORTOS ESSENCIAIS E A VIAGEM MÁXIMA

TEMPO

Devido a outras entregas de projetos e do início da época de exames achamos complicado a gestão de tempo