Gestão de Voos

Algoritmos e Estruturas de Dados

08:29AM

Enumeração de Classes

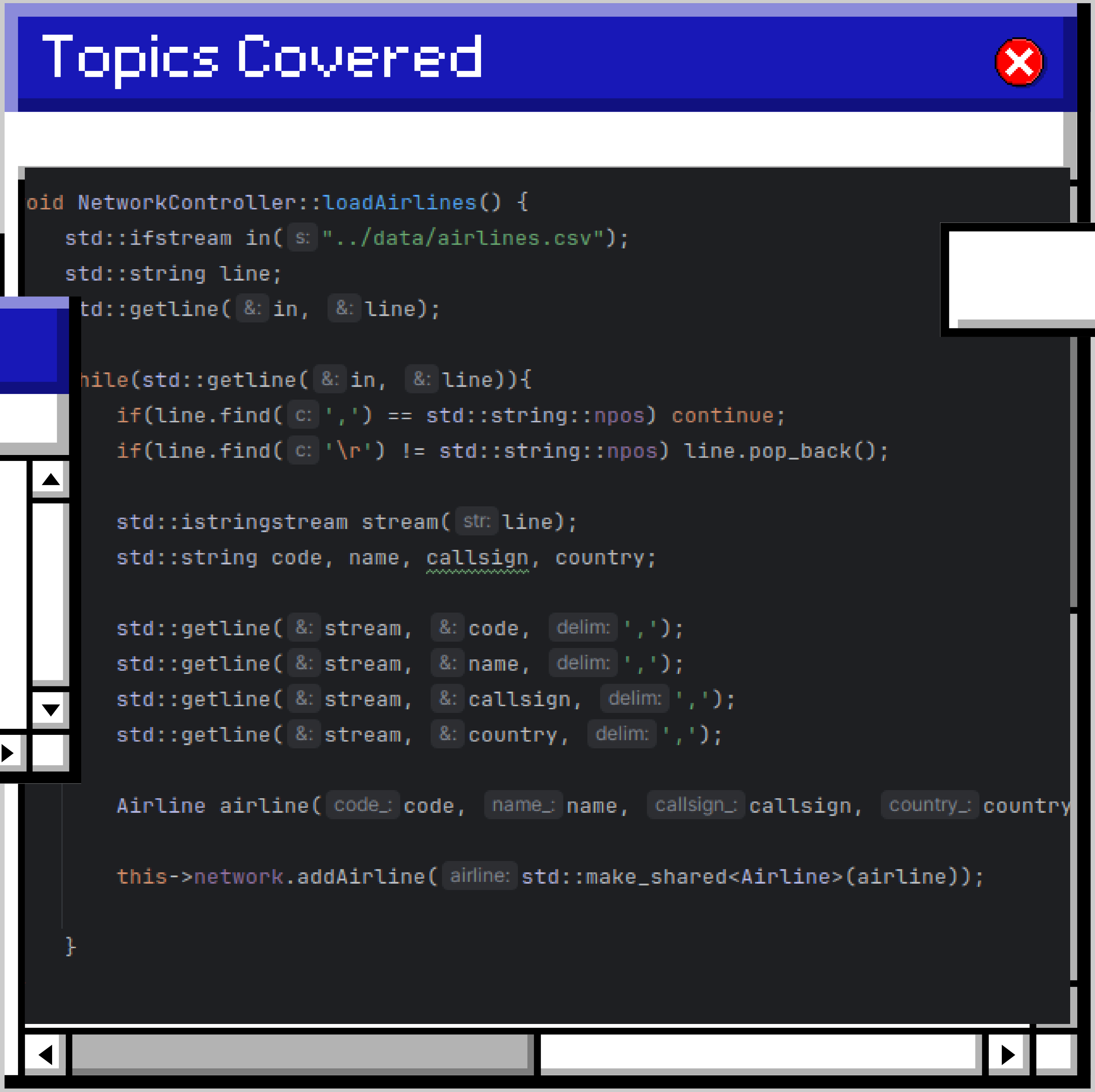
Classes	Atributos
Airline	code, name, callsign, country
Airport	code, name, city, country, latitude, longitude
AirportVertex	airport, flights, low, num, visited
SearchFilter	limitAirlines, filterAirlines, filterCities, airlinesToUse, citiesToStop
Flight	airlineCode, dest
Menu	controller, running
Network	flightCount, airportSet, airlineMap, cityMap
NetworkController	network

Voltar à página inicial

NetworkController

Network-Controller

Topics Covered



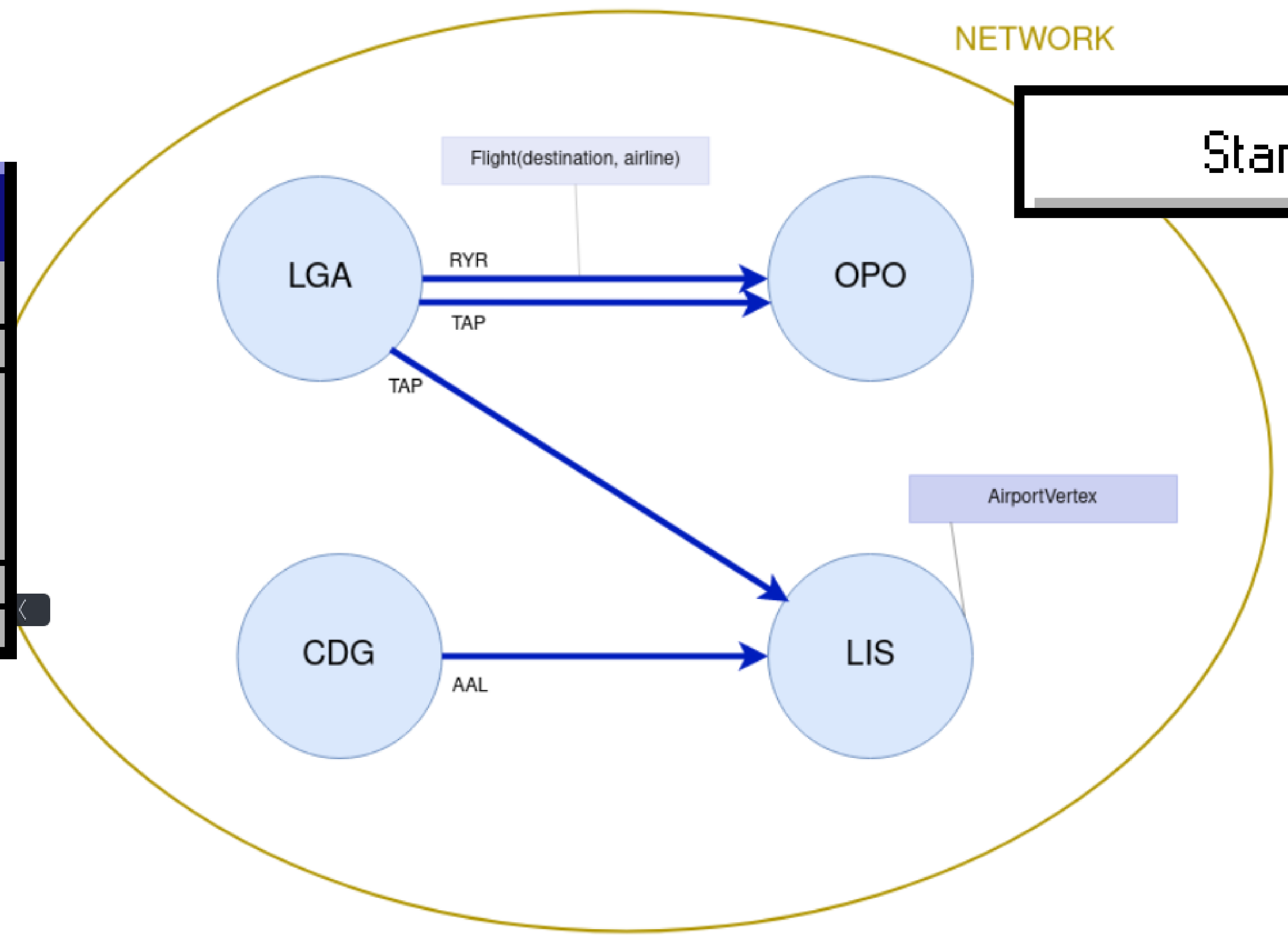
```
oid NetworkController::loadAirlines() {
   std::ifstream in( s: "../data/airlines.csv");
   std::string line;
    td::getline( &: in,  &: line);

    hile(std::getline( &: in,  &: line)){
       if(line.find( c: ',') == std::string::npos) continue;
       if(line.find( c: '\r') != std::string::npos) line.pop_back();

       std::istringstream stream( str: line);
       std::string code, name, callsign, country;

       std::getline( &: stream,  &: code,  delim: ',');
       std::getline( &: stream,  &: name,  delim: ',');
       std::getline( &: stream,  &: callsign,  delim: ',');
       std::getline( &: stream,  &: country,  delim: ',');

       Airline airline( code_: code,  name_: name,  callsign_: callsign,  country_: country

       this->network.addAirline( airline: std::make_shared<Airline>(airline));

   }
```

Start

Network

Topics Covered

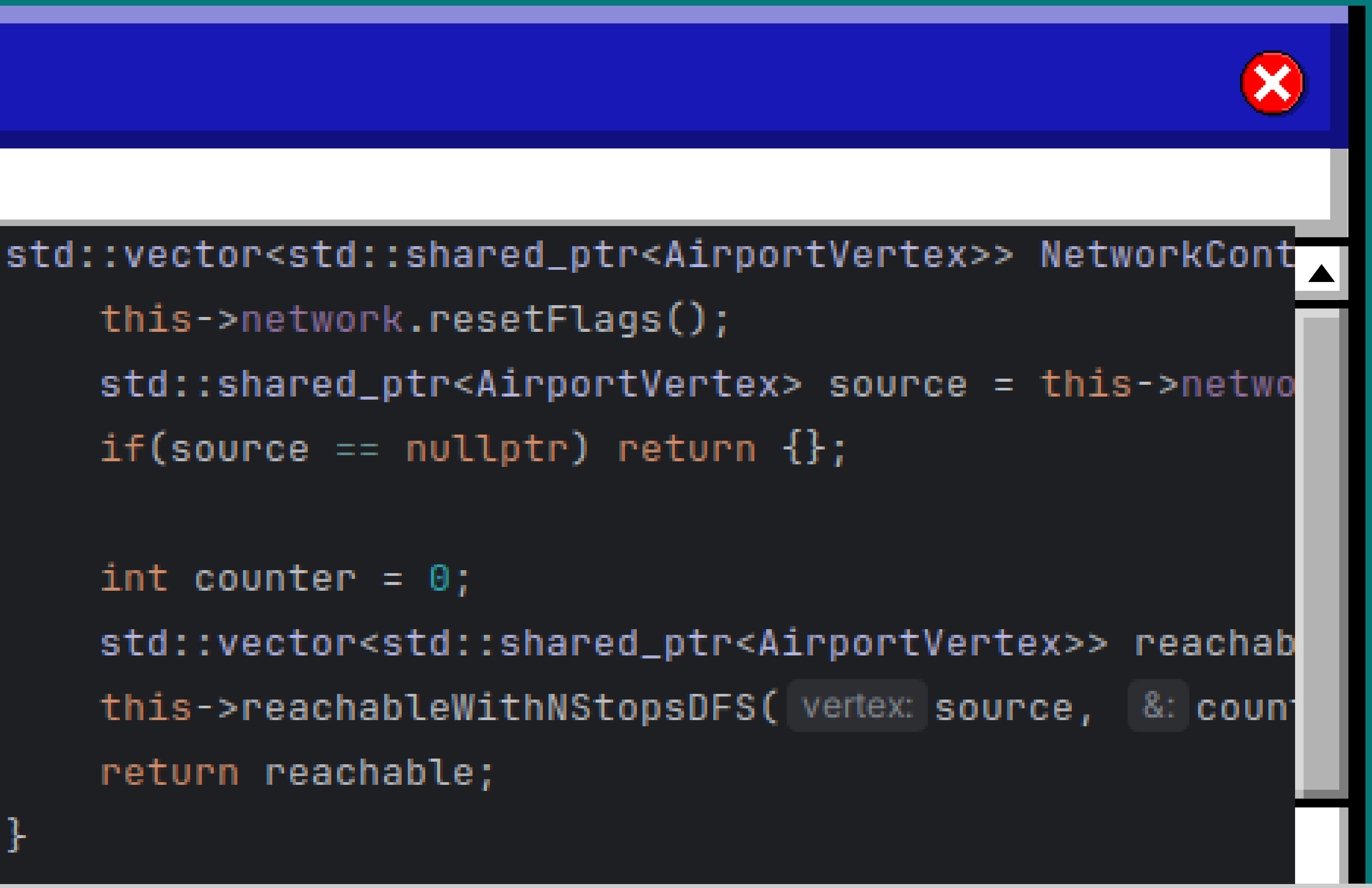
Network



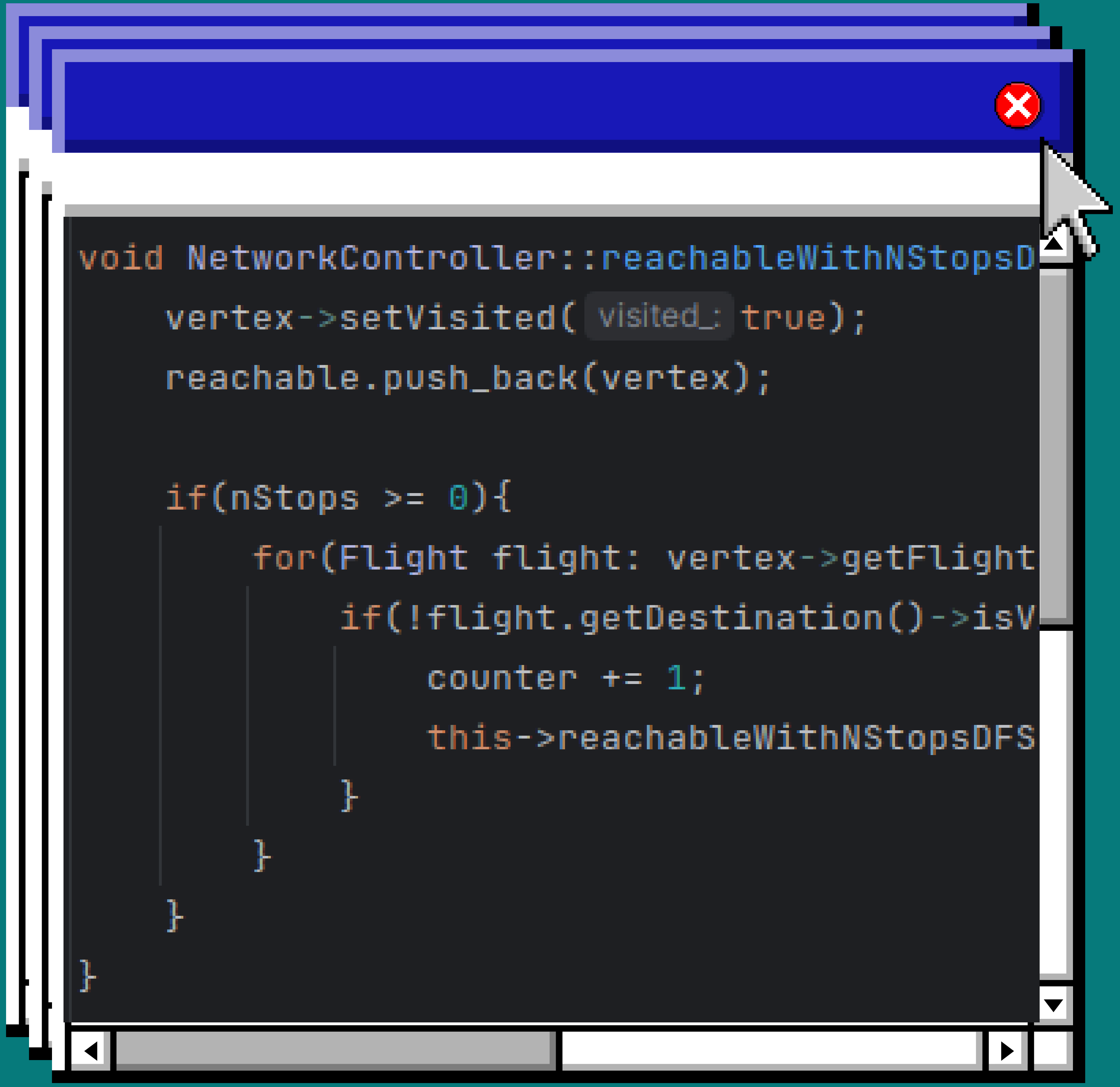
Start

reachableWithNStops

Complexidade: O(v + e)



```
std::vector<std::shared_ptr<AirportVertex>> NetworkCont
    this->network.resetFlags();
    std::shared_ptr<AirportVertex> source = this->netwo
    if(source == nullptr) return {};

    int counter = 0;
    std::vector<std::shared_ptr<AirportVertex>> reachab
    this->reachableWithNStopsDFS( vertex: source,  &: coun
    return reachable;
}
```

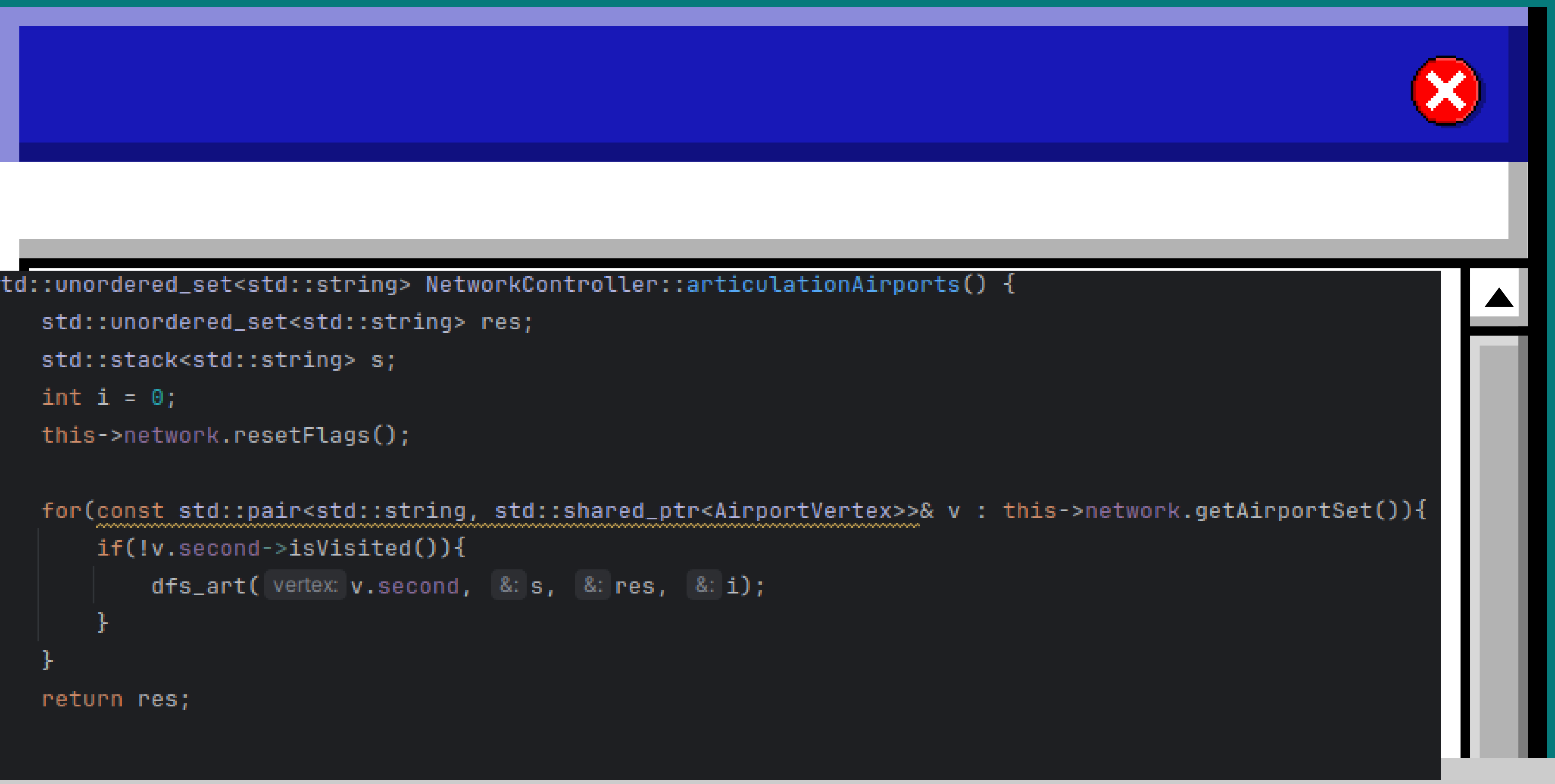


```
void NetworkController::reachableWithNStopsD
    vertex->setVisited( visited_: true);
    reachable.push_back(vertex);

    if(nStops >= 0){
        for(Flight flight: vertex->getFlight
            if(!flight.getDestination()->isV
                counter += 1;
                this->reachableWithNStopsDFS
            }
        }
    }
}
```

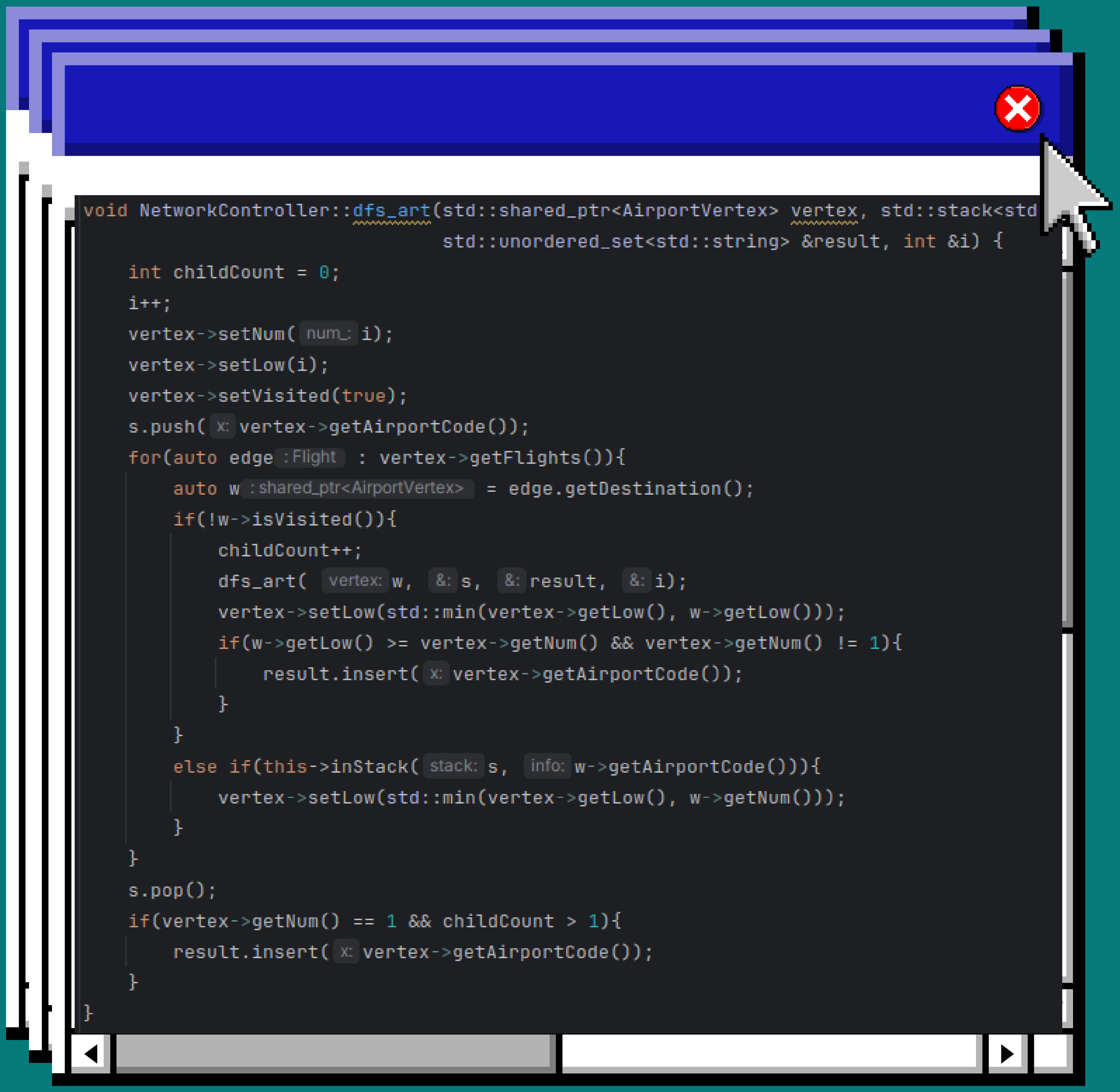
Back to Agenda Page

articulationAirports

Complexidade: O(v + e)



```
td::unordered_set<std::string> NetworkController::articulationAirports() {
    std::unordered_set<std::string> res;
    std::stack<std::string> s;
    int i = 0;
    this->network.resetFlags();

    for(const std::pair<std::string, std::shared_ptr<AirportVertex>>& v : this->network.getAirportSet()){
        if(!v.second->isVisited()){
            dfs_art( vertex: v.second,  &: s,  &: res,  &: i);
        }
    }
    return res;
```



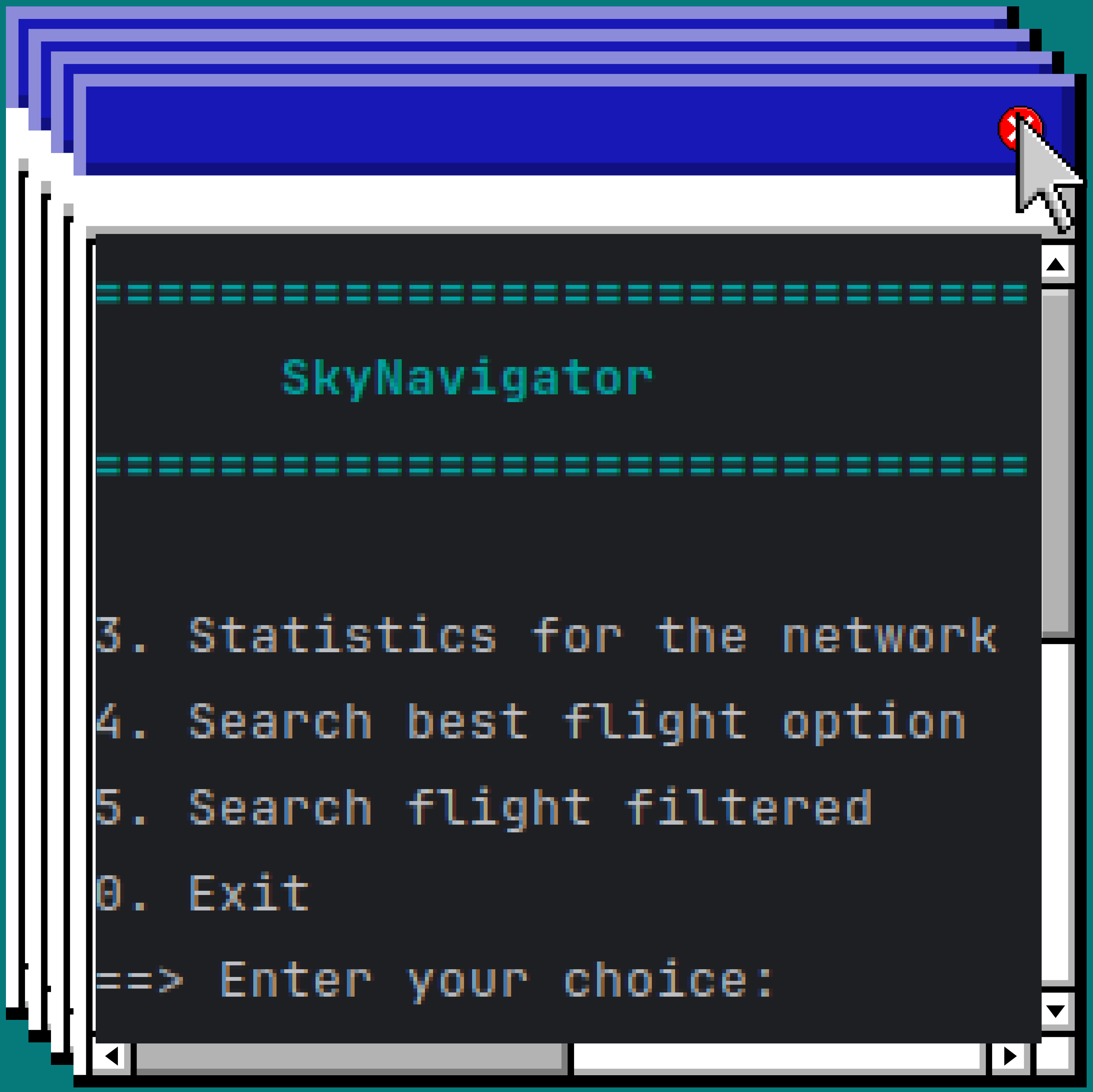
```
void NetworkController::dfs_art(std::shared_ptr<AirportVertex> vertex, std::stack<std
                                std::unordered_set<std::string> &result, int &i) {
    int childCount = 0;
    i++;
    vertex->setNum( num: i);
    vertex->setLow(i);
    vertex->setVisited(true);
    s.push( x: vertex->getAirportCode());
    for(auto edge :Flight : vertex->getFlights()){
        auto w :shared_ptr<AirportVertex> = edge.getDestination();
        if(!w->isVisited()){
            childCount++;
            dfs_art( vertex: w,  &: s,  &: result,  &: i);
            vertex->setLow(std::min(vertex->getLow(), w->getLow()));
            if(w->getLow() >= vertex->getNum() && vertex->getNum() != 1){
                result.insert( x: vertex->getAirportCode());
            }
        }
        else if(this->inStack( stack: s,  info: w->getAirportCode())){
            vertex->setLow(std::min(vertex->getLow(), w->getNum()));
        }
    }
    s.pop();
    if(vertex->getNum() == 1 && childCount > 1){
        result.insert( x: vertex->getAirportCode());
    }
}
```

Back to Agenda Page

Menu

Descrição

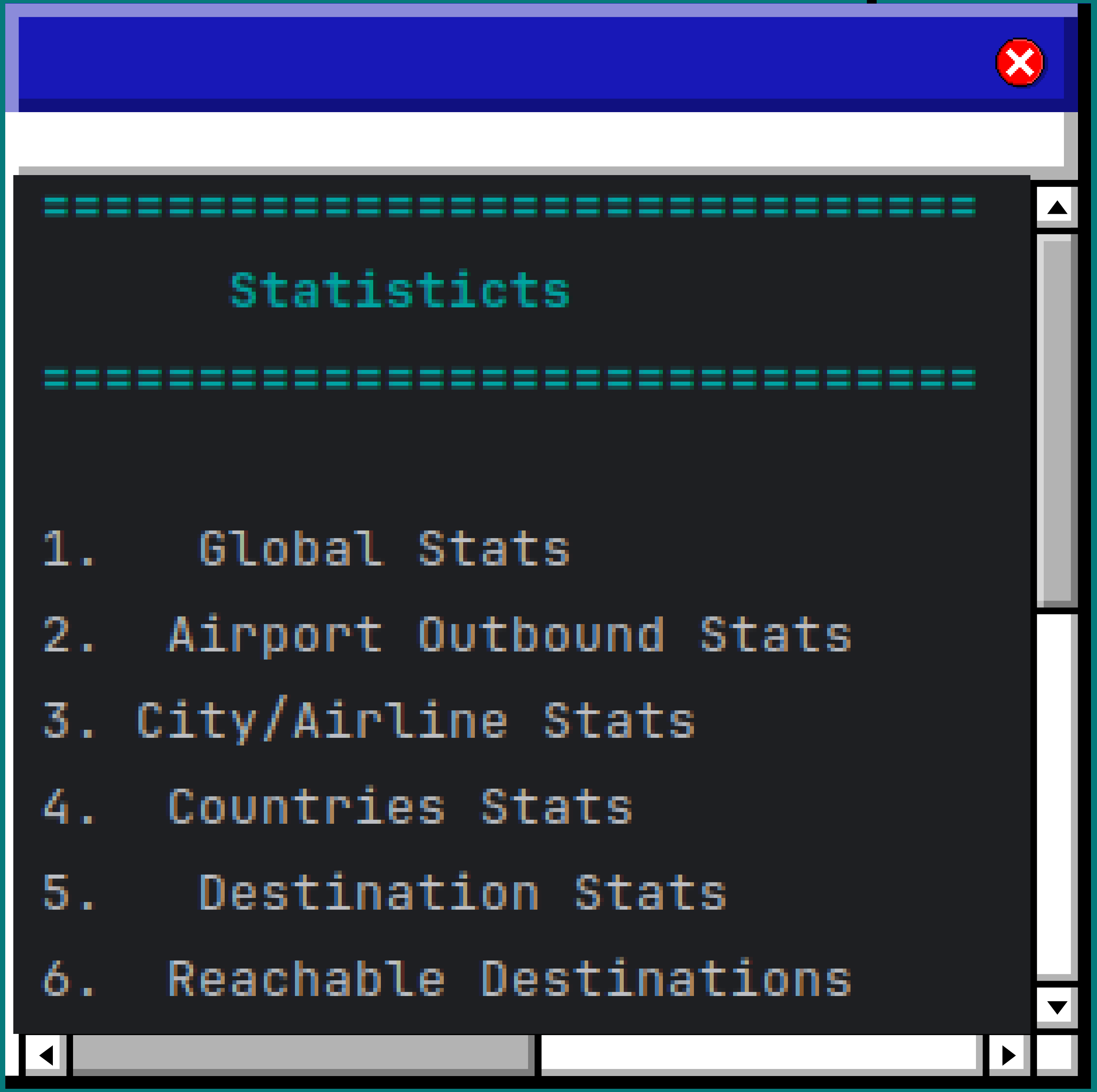
3. Dados da network
4. Escolher a melhor opção de voo
5. Filtrar o voo
0. Sair



```
==============================

      SkyNavigator

==============================


3. Statistics for the network
4. Search best flight option
5. Search flight filtered
0. Exit
==> Enter your choice:
```

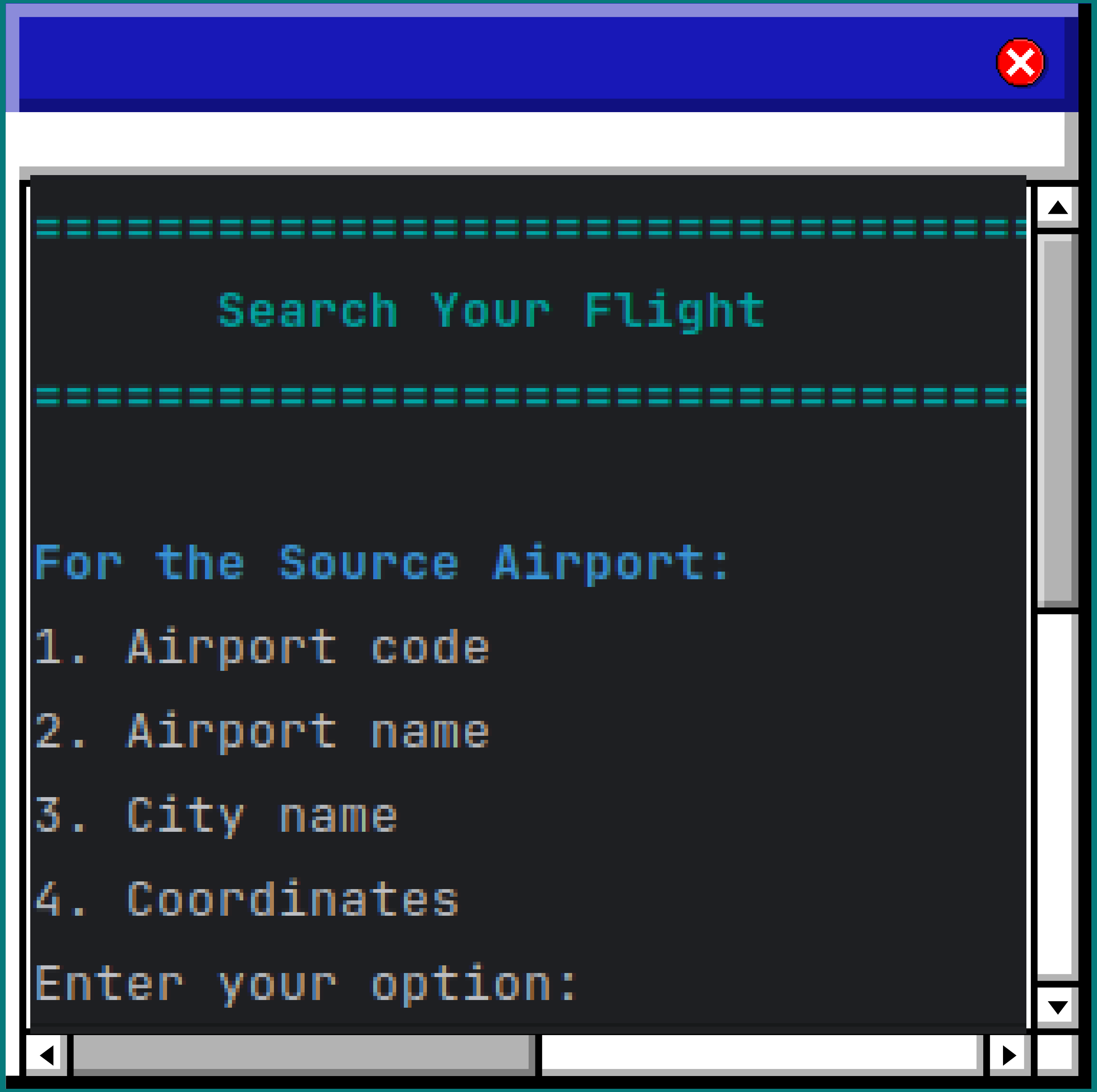
VOLTAR À PÁGINA INICIAL

Dados



```
==============================
      Statistcts
==============================


1.   Global Stats
2.  Airport Outbound Stats
3. City/Airline Stats
4.  Countries Stats
5.   Destination Stats
6.  Reachable Destinations
```

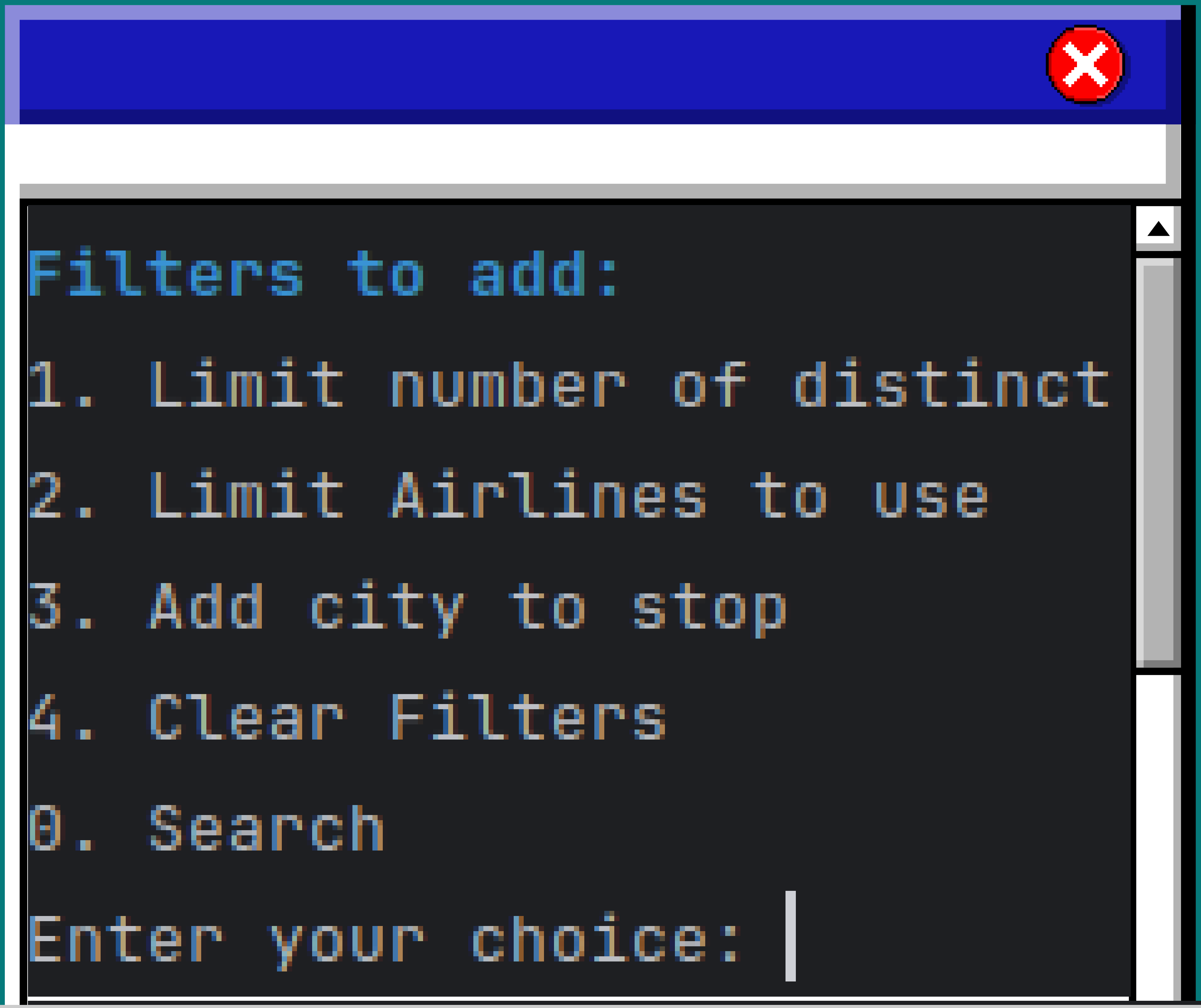
Pesquisar voo



```
=================================
      Search Your Flight
=================================


For the Source Airport:
1. Airport code
2. Airport name
3. City name
4. Coordinates
Enter your option:
```

VOLTAR À PÁGINA INICIAL

Filtros



```
Filters to add:
1. Limit number of distinct
2. Limit Airlines to use
3. Add city to stop
4. Clear Filters
0. Search
Enter your choice: 
```

VOLTAR À PÁGINA INICIAL

Filtros

Demonstração ao Vivo

Back to Agenda Page

FIM

Duarte Paiva Padrão -> 202206479

Nuno Miguel Oliveira Mota -> 202207549

Rodrigo Ferreira Alves -> 202207478