

Relatório EP0

Gabriel Henrique Pinheiro Rodrigues NUSP: 112.216-47

Março 2020

1 Como executar o EP na linha de comando:

Após compilado : ./ep0 T K C V S
onde:

- T - int - informa o tempo da simulação;
- K - int - número máximo de solicitações a torre de comando por unidade de tempo;
- C - int - número máximo de combustível;
- V - int - número máximo da duração do voo;
- S - int - semente da simulação;

2 Plano de Implementação

Na implementação do programa de gerenciamento de um aeroporto, foram utilizadas duas classes principais: A **classe Avião** e a **classe Aeroporto**, dentro da classe Aeroporto há 4 filas, cada uma separada pelo nível de prioridade (*). Também foram criados **2 structs**, um chamado **elementoDaFila**, que contém um ponteiro para o objeto da classe Avião e é utilizado como célula da fila, e outro struct chamado **cabecaTorreDeControle** que contém ponteiros para os elementos iniciais e finais de cada uma das 4 filas;

(*) As **prioridades** foram definidas da seguinte forma:

- 3 - Prioridade máxima - Corresponde aos aviões sem combustível;
- 2 - Voos de emergência - Ex. presidencial, acidentes;
- 1 - Aviões que já estão esperando por mais de 10% do tempo total de voo para decolar;
- 0 - Demais aviões;

A **classe Avião** contém:

- O nome da companhia aérea;
- O número do avião;
- Se a intenção é pousar ou decolar;
- Se é um voo de emergência ou não;
- O nome do aeroporto de origem ou destino;
- A quantidade de combustível;
- A duração do voo;
- O tempo de espera;
- A prioridade do voo;

A **classe Aeroporto** contém:

- O número de elementos da fila;
- A disponibilidade de cada uma das 3 pistas;
- Ponteiro para a cabeça `TorreDeControle` - Struct que contém ponteiros que apontam para os elementos iniciais e finais de cada uma das 4 filas;
- Métodos para a manipulação da fila;
- Métodos para a atualização da disponibilidade das pistas;
- Métodos de exibição dos elementos da fila;

Por motivos de encapsulamento os elementos da fila são estruturas que apontam para aviões, essas estruturas também possuem o endereço do próximo elemento da fila e para o anterior, dessa forma a fila implementada é uma lista duplamente encadeada;

3 Funcionamento do Aeroporto

A cada unidade de tempo os métodos da classe `Aeroporto` que são executados são:

- **`contatoComATorre(Aviao* aviao)`** - Verifica se é possível inserir o avião em alguma fila ou se é melhor que esse seja direcionado a outro aeroporto (*1);
- **`liberaVoos()`** - Enquanto houver pistas disponíveis tenta liberar os aviões para pouso e decolagem respeitando os níveis de prioridade;

- **atualizaSituacaoDosAvioesNaFila()** - Decrementa a quantidade de combustível dos aviões na fila, aumenta o tempo de espera e altera a prioridade caso necessário. (Caso algum avião de prioridade 0 fica sem combustível, sua prioridade vai para 3, se o avião já estiver esperando por mais de 10% do seu tempo de voo, sua prioridade vai para 1);
- **atualizaDispDePista()** - Decrementa o tempo de espera para a liberação das pistas e troca a sua disponibilidade;
- **coletaEstatisticasEPrinta()** - Armazena dados interessantes e exibe na hora da execução - Ex. número de aviões que caíram, número de aviões enviados a outros aeroportos, ...

(*1) Critérios utilizados:

- Se já há 9 emergências na fila e o avião atual é de pouso de emergência esse será enviado a outro aeroporto;
- Se ainda não há 9 emergências na fila e o avião atual é de pouso de emergência, mas sua quantidade de combustível é menor que o tempo que os outros aviões de emergência vão levar para liberar a pista, então ele será enviado a outro aeroporto;
- Se o avião tem prioridade 0, mas sua quantidade de combustível é menor que o tempo de liberação da pista, então ele também é enviado a outro aeroporto;
- Caso contrário, ele é inserido em sua respectiva fila(De acordo com a sua prioridade);

4 Testes

Para gerar os objetos da classe Avião criamos uma função que monta de maneira semi-aleatória o objeto, a **proporção de voos de emergência é 10%**, foram escolhidas **7 companhias aéreas** e **33 aeroportos**, a probabilidade de um avião querer pousar ou decolar é a mesma.

4.1 Os testes realizados foram

0-) INPUT: T=500, K=3, C=20, V=20, S=17

OUTPUT:

- Os aviões que estão esperando para pousar são: Nenhum;
- Os aviões que estão esperando para decolar são: Nenhum;

- A quantidade média de combustível dos aviões que pousaram: 10.7351;
- A quantidade de aviões pousando/decolando em condições de emergência é: 29;
- A quantidade total de aviões que já pousaram é: 185;
- A quantidade total de aviões que já decolaram é: 201;
- A quantidade de aviões que caíram é: 0;
- A porcentagem de aviões enviados para outros aeroportos é: 3%;

1-) INPUT: T=500, K=5, C=20, V=20, S=17

OUTPUT:

- Os aviões que estão esperando para pousar são: Nenhum;
- Os aviões que estão esperando para decolar são: Nenhum;
- A quantidade média de combustível dos aviões que pousaram: 9.94595;
- A quantidade de aviões pousando/decolando em condições de emergência é: 39;
- A quantidade total de aviões que já pousaram é: 296;
- A quantidade total de aviões que já decolaram é: 270;
- A quantidade de aviões que caíram é: 0;
- A porcentagem de aviões enviados para outros aeroportos é: 12%;

2-) INPUT: T=500, K=7, C=20, V=20, S=17

OUTPUT:

- Os aviões que estão esperando para pousar são: LA465; AZ207; AZ761;
- Os aviões que estão esperando para decolar são: AZ207; AZ761;
- A quantidade média de combustível dos aviões que pousaram: 4.19298;
- A quantidade de aviões pousando/decolando em condições de emergência é: 36;
- A quantidade total de aviões que já pousaram é: 228;
- A quantidade total de aviões que já decolaram é: 343;
- A quantidade de aviões que caíram é: 0;
- A porcentagem de aviões enviados para outros aeroportos é: 44%;