

Trabalho redigido por Gabriel Haruo Hanai Takeuchi, de NUSP 13671636.

Introdução

Este arquivo `README.md` oferece uma documentação para a minha interpretação do EP1 de MAC0323. Arquivos inclusos no pacote: `main.cpp`, `aerolib.hpp`, `aerolib.cpp`, `README.md`, `README.pdf`.

Como compilar e executar

```
clear && g++ -Wall -std=c++11 *.cpp -o a.out && ./a.out
```

Os parâmetros podem ser modificados de acordo com a primeira parte da sessão [Testes](#)

Classes e objetos

Duas classes foram usadas: `class Pista` e `class Aviao`, definidas a seguir:

```
class Pista {  
public:  
    int livre;  
  
    Pista();  
  
    void cooldown();  
};
```

Observação: Uma pista livre assume o valor `livre == 1`. Quando recém-ocupada, assume o valor `livre == -2`. É importante ressaltar que, para simular o cooldown das pistas, `livre` é incrementado a cada rodada até assumir novamente o valor `livre == 1`.

```
class Aviao {  
public:  
    string id, codigo;  
    int prioridade, contador, combustivel, tempoVoo;  
    // se pouso, tempoVoo = -1  
    // se decolagem, combustivel = -1  
    bool ehPouso;  
  
    Aviao(float Pp, float Pe, bool ehPouso, int C, int V);  
  
    string setId();  
    string setCodigo(bool ehPouso);  
    int setCombustivel(int C);  
    int setTempoVoo(int V);
```

```
int setPrioridade(float Pe, int C, int V);  
};
```

Observação: A propriedade `contador` contabiliza quanto tempo um avião está na fila de prioridade. Útil para calcular o combustível restante de um avião ou o seu tempo de voo. A propriedade `prioridade` será explicada em breve [nesta sessão](#).

Estruturas de dados usadas

Foram usadas, no total, 7 estruturas de dados:

- uma fila de prioridade
- uma lista ligada para monitorar os aviões que ainda não pousaram
- uma lista ligada para monitorar os aviões que ainda não decolaram
- uma lista ligada para monitorar os aviões que já pousaram
- uma lista ligada para monitorar os aviões que já decolaram
- uma lista ligada para monitorar os pousos que caíram
- um vetor para armazenar as pistas de voo

```
priority_queue<Aviao*, vector<Aviao*>, decltype(comparaAviao)>  
queue(comparaAviao);  
list<Aviao*> pousosNaQueue;  
list<Aviao*> decolagensNaQueue;  
list<Aviao*> pousosForaQueue;  
list<Aviao*> decolagensForaQueue;  
list<Aviao*> avioesMortos;  
vector<Pista> pistas;
```

Listas ligadas auxiliares

As listas ligadas foram usadas para ter maior controle e organização de todos os aviões gerados, visto que não é possível iterar sobre a `std::priority_queue`. Além disso, a separação em listas menores facilita a construção das estatísticas requisitadas no enunciado (tempo médio de espera, quantidade média de combustível, % de acidentes, etc.)

```
list<Aviao*> pousosForaQueue;  
list<Aviao*> decolagensForaQueue;  
list<Aviao*> pousosNaQueue;  
list<Aviao*> decolagensNaQueue;  
list<Aviao*> avioesMortos;
```

A fila de prioridade

```
priority_queue<Aviao*, vector<Aviao*>, decltype(comparaAviao)>
queue(comparaAviao);
```

Como requisitado no enunciado, a estrutura `std::priority_queue` customizada armazena ponteiros para os objetos `Aviao*`. Um detalhe importante é o comparador `comparaAviao`, o qual efetivamente torna a fila em um min heap de acordo com a `prioridade` dos aviões.

Prioridade de um avião

A propriedade `int prioridade` da classe `Aviao` toma

- (valor 0) caso o avião seja gerado como uma emergência
- (valor do combustível + número da rodada atual) caso seja um pouso
- (valor do tempo máximo de voo + número da rodada atual) caso seja uma decolagem

A `prioridade` é determinada na geração de um avião, e permanece estática ao longo do algoritmo. Os aviões que tiverem o menor valor absoluto de `prioridade` serão tratados com mais urgência. A soma com o 'número da rodada atual' significa que o avião aguenta ficar na fila até a rodada $n = (\text{combustível ou tempo máximo} + \text{número da rodada atual})$.

Definição de 'emergência'

Em primeira instância, uma emergência é definida no momento de geração de um avião com probabilidade `Pe` - nesse caso, `prioridade` é definida como 0. Entretanto, aglutina-se nessa definição as situações em que o combustível de um avião chega a 0 e quando o tempo de espera ultrapassa 10% do limite.

Testes

Os testes podem ser reproduzidos mudando a seed e as entradas em `main.cpp` (comentando ou descomentando as linhas certas para mudar de modo interativo para modo automático):

```
// em main.cpp

int seed = 123;

// modo interativo
// cout << "Digite T, K, Pp, Pe, C e V, respectivamente: ";
// cin >> T >> K >> Pp >> Pe >> C >> V;

// modo automático
T = 10; K = 2; Pp = 0.5; Pe = 0.1; C = 5; V = 5;
```

Todos os exemplos de testes são da última rodada, em que as estatísticas estão completas. As estatísticas consideradas foram:

1. Tempo médio de espera para pousos finalizados

2. Tempo médio de espera para decolagens finalizadas
3. Quantidade média de combustível para pousos não-finalizados
4. Quantidade média de combustível para pousos finalizados
5. Quantidade total de aviões gerados
6. Quantidade de emergências (gerado como emergência) desviadas
7. Quantidade de emergências (combustível 0, tempo de voo acima do teto) desviadas
8. Quantidade de aviões que caíram

Teste 01

T pequeno, K pequeno, Pp == Pd, Pe pequeno, C médio, V médio.

```
T = 10; K = 2; Pp = 0.5; Pe = 0.1; C = 5; V = 5;
```

```
int seed = 123;
```

```
// output
```

```
RELATÓRIO DA RODADA:
```

```
Pistas: -2 | -2 | -2
```

```
Avião BU445 esperando para pousar com combustível atual 0 e prioridade 10
```

```
Avião IB418 esperando para pousar com combustível atual 4 e prioridade 14
```

```
Avião YQ056 esperando para decolar com tempo de voo atual 1 e prioridade 10
```

```
Avião DZ240 esperando para decolar com tempo de voo atual 0 e prioridade 12
```

```
Tempo médio de espera (pousos): 1.75
```

```
Tempo médio de espera (decolagens): 1.83333
```

```
Quantidade média de combustível (esperando pousar): 2
```

```
Quantidade média de combustível (já pousaram): 2.75
```

```
Quantidade total de aviões gerados: 16
```

```
Quantidade de emergências (gerado como emergência) desviadas: 0 (0%)
```

```
Quantidade de emergências (outras) desviadas: 2 (12.5%)
```

```
Quantidade de aviões que caíram: 0 (0%)
```

```
int seed = 456;
```

```
// output
```

```
RELATÓRIO DA RODADA:
```

```
Pistas: -2 | -2 | 1
```

```
Avião UD881 esperando para pousar com combustível atual 1 e prioridade 11
```

```
Avião GN788 esperando para pousar com combustível atual 2 e prioridade 12
```

```
Avião SN091 esperando para pousar com combustível atual 1 e prioridade 11
```

```
Avião IT652 esperando para pousar com combustível atual 5 e prioridade 15
```

```
Tempo médio de espera (pousos): 1.25
```

```
Tempo médio de espera (decolagens): 1.16667
```

```
Quantidade média de combustível (esperando pousar): 2.25
Quantidade média de combustível (já pousaram): 1.25
Quantidade total de aviões gerados: 15
Quantidade de emergências (gerado como emergência) desviadas: 0 (0%)
Quantidade de emergências (outras) desviadas: 1 (6.66667%)
Quantidade de aviões que caíram: 0 (0%)
```

```
int seed = 789;
```

```
// output
```

```
RELATÓRIO DA RODADA:
```

```
Pistas: -2 | -2 | -1
```

```
Avião ZJ822 esperando para decolar com tempo de voo atual 1 e prioridade 14
```

```
Avião DU700 esperando para decolar com tempo de voo atual 0 e prioridade 15
```

```
Avião PX142 esperando para decolar com tempo de voo atual 0 e prioridade 15
```

```
Tempo médio de espera (pousos): 3.75
```

```
Tempo médio de espera (decolagens): 1
```

```
Quantidade média de combustível (esperando pousar): -nan
```

```
Quantidade média de combustível (já pousaram): 0.5
```

```
Quantidade total de aviões gerados: 17
```

```
Quantidade de emergências (gerado como emergência) desviadas: 1 (5.88235%)
```

```
Quantidade de emergências (outras) desviadas: 3 (17.6471%)
```

```
Quantidade de aviões que caíram: 0 (0%)
```

Com esses parâmetros, observa-se que o algoritmo é capaz de lidar com as demandas com certa eficiência:

- Em média, 2.25 tempos para pouso e 1.5 tempos para decolagem
- Média taxa de aviões desviados (em média, 12.2712% do total de voos)
- Baixa taxa de quedas (em média, 0 aviões caíram).

Teste 02

T grande, K pequeno, Pp == Pd, Pe pequeno, C médio, V médio.

```
T = 1000; K = 2; Pp = 0.5; Pe = 0.1; C = 5; V = 5;
```

```
int seed = 123;
```

```
// output
```

```
Tempo médio de espera (pousos): 6.32616
```

```
Tempo médio de espera (decolagens): 2.82231
```

```
Quantidade média de combustível (esperando pousar): 3
```

```
Quantidade média de combustível (já pousaram): 1.17563
```

```
Quantidade total de aviões gerados: 1506
```

```
Quantidade de emergências (gerado como emergência) desviadas: 70 (4.64807%)
```

```
Quantidade de emergências (outras) desviadas: 452 (30.0133%)
Quantidade de aviões que caíram: 64 (4.24967%)
```

```
int seed = 456;

// output
Tempo médio de espera (pousos): 6.92857
Tempo médio de espera (decolagens): 2.82
Quantidade média de combustível (esperando pousar): 3
Quantidade média de combustível (já pousaram): 1.34127
Quantidade total de aviões gerados: 1498
Quantidade de emergências (gerado como emergência) desviadas: 70 (4.6729%)
Quantidade de emergências (outras) desviadas: 462 (30.8411%)
Quantidade de aviões que caíram: 58 (3.87183%)
```

```
int seed = 789;

// output
Tempo médio de espera (pousos): 6.15926
Tempo médio de espera (decolagens): 2.74744
Quantidade média de combustível (esperando pousar): 1
Quantidade média de combustível (já pousaram): 1.3
Quantidade total de aviões gerados: 1491
Quantidade de emergências (gerado como emergência) desviadas: 69 (4.62777%)
Quantidade de emergências (outras) desviadas: 448 (30.0469%)
Quantidade de aviões que caíram: 42 (2.8169%)
```

Com um **T** grande, o algoritmo ainda se mantém consistente:

- Em média, 6.4633 tempos para pouso (aumento de 2.87 vezes em relação ao teste 01) e 2.7965 tempos para decolagem (aumento de 1.86 vezes em relação ao teste 01)
- Alta taxa de emergências desviadas (em média, 30.3%, aumento de 2.4691 vezes em relação ao teste 01)
- Baixa taxa de aviões caídos (em média, 3.6461%)

Teste 03

T grande, **K** grande, **Pp** == **Pd**, **Pe** pequeno, **C** grande, **V** grande.

```
T = 1000; K = 10; Pp = 0.5; Pe = 0.1; C = 10; V = 10;
```

```
int seed = 123;

// output
RELATÓRIO DA RODADA:
```

```

Tempo médio de espera (pousos): 41.9916
Tempo médio de espera (decolagens): 5.93857
Quantidade média de combustível (esperando pousar): 2.9
Quantidade média de combustível (já pousaram): 11.2185
Quantidade total de aviões gerados: 5707
Quantidade de emergências (gerado como emergência) desviadas: 319 (5.58963%)
Quantidade de emergências (outras) desviadas: 2689 (47.1176%)
Quantidade de aviões que caíram: 1712 (29.9982%)

```

```

int seed = 456;

// output
Tempo médio de espera (pousos): 33.1985
Tempo médio de espera (decolagens): 5.92755
Quantidade média de combustível (esperando pousar): 3.5
Quantidade média de combustível (já pousaram): 10.7059
Quantidade total de aviões gerados: 5421
Quantidade de emergências (gerado como emergência) desviadas: 334 (6.16122%)
Quantidade de emergências (outras) desviadas: 2502 (46.1538%)
Quantidade de aviões que caíram: 1592 (29.3673%)

```

```

int seed = 789;

// output
Tempo médio de espera (pousos): 31.0657
Tempo médio de espera (decolagens): 5.875
Quantidade média de combustível (esperando pousar): 4.05556
Quantidade média de combustível (já pousaram): 10.4015
Quantidade total de aviões gerados: 5535
Quantidade de emergências (gerado como emergência) desviadas: 311 (5.61879%)
Quantidade de emergências (outras) desviadas: 2514 (45.4201%)
Quantidade de aviões que caíram: 1708 (30.8582%)

```

Com um **T** grande e **K** grande, observa-se:

- Em média, 35.4186 tempos para pouso e 5.9137 tempos para decolagem
- Altíssima taxa de emergências desviadas (em média, 46.2305%)
- Alta taxa de aviões caídos (em média, 30.0745%) Esses resultados já eram esperados, visto que temos um **K** excessivo.

Teste 04

T grande, **K** pequeno, **Pp** == **Pd**, **Pe** absoluto, **C** médio, **V** médio.

```
T = 1000; K = 2; Pp = 0.5; Pe = 1.0; C = 5; V = 5;
```

```
int seed = 123;

// output
Tempo médio de espera (pousos): 0
Tempo médio de espera (decolagens): 0
Quantidade média de combustível (esperando pousar): -nan
Quantidade média de combustível (já pousaram): 3.20475
Quantidade total de aviões gerados: 1506
Quantidade de emergências (gerado como emergência) desviadas: 628 (41.6999%)
Quantidade de emergências (outras) desviadas: 0 (0%)
Quantidade de aviões que caíram: 0 (0%)
```

```
int seed = 456;

// output
Tempo médio de espera (pousos): 0
Tempo médio de espera (decolagens): 0
Quantidade média de combustível (esperando pousar): -nan
Quantidade média de combustível (já pousaram): 3.49379
Quantidade total de aviões gerados: 1498
Quantidade de emergências (gerado como emergência) desviadas: 623 (41.5888%)
Quantidade de emergências (outras) desviadas: 0 (0%)
Quantidade de aviões que caíram: 0 (0%)
```

```
int seed = 789;

// output
Tempo médio de espera (pousos): 0
Tempo médio de espera (decolagens): 0
Quantidade média de combustível (esperando pousar): -nan
Quantidade média de combustível (já pousaram): 3.3038
Quantidade total de aviões gerados: 1491
Quantidade de emergências (gerado como emergência) desviadas: 606 (40.6439%)
Quantidade de emergências (outras) desviadas: 0 (0%)
Quantidade de aviões que caíram: 0 (0%)
```

Com $Pe == 1.0$ e K pequeno, observa-se que a média de emergências desviadas se concentra em 41.3108%, ou seja, mais da metade dos voos foram posicionados nas pistas. Um resultado otimista.

Conclusão

Infelizmente, esse algoritmo não é o mais eficiente nem o mais elegante, mas se mostra minimamente consistente em relação às quedas de avioes. A experiência de ter um primeiro contato com C++ e tentar construir um sistema foi enriquecedora. Aguardo os feedbacks.