

## MAC 122 - Princípios de Desenvolvimento de Algoritmos

Segundo semestre de 2015

Quarto Exercício-Programa – Entrega: **9 de novembro**

### Controlando um aeroporto

Neste exercício-programa vamos simular o funcionamento de uma torre de controle de um aeroporto comercial. Nosso aeroporto conta com duas pistas: uma principal, usada para pousos e decolagens, e uma pista auxiliar. A torre de controle recebe requisições de pousos e decolagens das aeronaves com as seguintes informações:

`<tempo atual> P/D <id do voo> <ind de prioridade> <tempo para atendimento>`

- `tempo atual` indica o momento em que a requisição chegou;
- `P/D` identifica se é uma requisição de pouso ou decolagem;
- `id do voo` é o código de identificação do voo;
- `ind de prioridade` tem valor 0 se é um voo comercial normal ou 1 se o voo é uma emergência médica, avião de autoridade ou outra emergência qualquer;
- `tempo para atendimento` indica o tempo máximo que o avião pode permanecer com segurança aguardando o pouso ou a decolagem

Além disso, a torre de controle pode receber notificações do serviço meteorológico, indicando que o aeroporto deve ser fechado por conta de condições meteorológicas desfavoráveis ou se pode ser reaberto. Isso é indicado por uma requisição iniciada pelo `<tempo atual>` seguido pela letra **F** (fechado) ou **A** (aberto). No início do dia nenhum pouso ou decolagem é realizada antes que o serviço meteorológico autorize a abertura do aeroporto. Observe que um voo normal pode se tornar uma emergência caso ele esteja na fila para pousar e o aeroporto ficar fechado por muito tempo. O fim da operação do aeroporto é indicado por uma requisição contendo a letra **Z**.

Sua tarefa será escrever um algoritmo que organize as filas de pouso e decolagem. Estas filas devem ser implementadas utilizando listas ligadas a fim de que seja eficiente remover um elemento qualquer da fila. Em cada unidade de tempo, além de receber as requisições descritas acima, seu algoritmo poderá decidir pela utilização da pista principal (se o aeroporto estiver aberto) ou, em um caso excepcional, pela pista auxiliar. Neste período deve atender as requisições e, na medida do possível, não perder nenhum avião ;) A ação a ser realizada deve ser indicada através de uma mensagem:

`<tempo> Autorizado pouso/decolagem de <id do voo> na pista principal/auxiliar`

### Exemplo de entrada

```
1 P JJ3124 0 20
2 D G31500 0 15
3 P AZ1234 0 20
4 D USF001 1 5
5 A
6 P VR2301 0 10
7 F
10 P DT0123 0 7
13 A
15 D SUS001 1 5
25 Z
```

### Exemplo de saída

Para a entrada acima, uma possível saída seria dada por:

```
5 Aeroporto aberto
6 Autorizado decolagem de USF001 na pista principal
7 Aeroporto fechado
13 Aeroporto aberto
14 Autorizado pouso de VR2301 na pista principal
15 Autorizado pouso de JJ3124 na pista principal
16 Autorizado decolagem de SUS001 na pista principal
17 Autorizado decolagem de G31500 na pista principal
17 Autorizado pouso de DT0123 na pista auxiliar
18 Autorizado pouso de AZ1234 na pista principal
25 Aeroporto fechado
```

Esperamos que neste exercício você também entregue um relatório em que descreve o algoritmo utilizado para resolver o problema, que estruturas de dados você utilizou, os testes que realizou, suas conclusões, etc. Pense em métricas que possam mostrar que as decisões que você tomou foram boas, por exemplo, calcule o tempo médio (máximo) de permanência na fila de pouso/decolagem, número de aviões “perdidos” durante a simulação, número de vezes que a pista auxiliar teve de ser utilizada, etc. Muito material na Internet pode ser encontrado com respeito a tráfego aéreo que pode ser útil para gerar os dados de teste.