

## **Exercícios: estrutura de dados Fila**

### **Questão 1.**

Implemente um sistema de gerenciamento de fila de banco. O programa deve apresentar um menu ao usuário com as seguintes opções:

- Atendimento normal
- Atendimento preferencial
- Mostrar senhas chamadas até o momento

O sistema fará o gerenciamento de duas filas (normal e preferencial). Quando um usuário selecionar um tipo de atendimento o sistema exibirá a senha do usuário e adicionará esta senha a uma das filas.

Quando a opção de mostrar as senhas for selecionada o sistema irá realizar o desenfileiramento das senhas nas duas filas, mas obedecendo às regras a baixo:

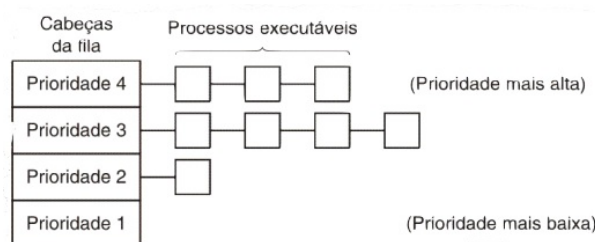
- Desenfileire todas as senhas nas duas filas, exceto à ultima gerada
- As senhas são chamadas intercalando senhas das duas filas
- Se houver senhas na fila prioritária ela deve ser chamada primeiro
- Se não houver senhas na fila prioritária a próxima senha a ser chamada será da fila normal

### **Questão 2.**

Nos sistemas multi programados onde existem vários processos em execução que compartilham recursos e todos precisam serem executados pelo processador, é fundamental a gerência desse recurso pelo sistema operacional.

A partir do momento em que vários processos podem estar prontos para executar, critérios devem ser estabelecidos para determinar qual processo será escolhido para fazer uso do processador.

Escalonamento por Prioridades: neste escalonamento é feita a preempção dos processos em execução na prioridade de cada processo. Este escalonamento foi projetado para sistemas de tempo real. Nele o processo com maior prioridade que está pronto para a execução é sempre o escolhido para execução, e processos com valores iguais são escalonados seguindo o critério FIFO. O Sistema Operacional



atende inicialmente as filas de prioridade mais alta, então apenas quando uma fila é esvaziada é que o escalonador passa para a próxima fila.

Crie uma estrutura de dados Fila de Prioridades cujo funcionamento simule o escalonamento de processos por prioridades e demonstre o seu funcionamento.

### Questão 3.

Escreva um programa que simule o controle de uma pista de decolagem de aviões em um aeroporto. Neste programa, o usuário deve ser capaz de realizar as seguintes tarefas:

- a) Listar o número de aviões aguardando na fila de decolagem;
- b) Autorizar a decolagem do primeiro avião da fila;
- c) Adicionar um avião à fila de espera;
- d) Listar todos os aviões na fila de espera;
- e) Listar as características do primeiro avião da fila.

Considere que os aviões possuem um nome e um número inteiro como identificador. Adicione outras características conforme achar necessário.

### Questão 4.

Crie uma estrutura Pessoa para guardar informações de nome e idade. Carregue as informações das pessoas contidas no arquivo name\_age.txt.

Carregue as pessoas do arquivo em uma fila, em seguida use apenas as operações *push* e *pop* e variáveis ou filas adicionais para ordenar a fila inicial pela idade da maior idade para a menor (mais velhos primeiro).

### Questão 5.

Dado um número  $n$ , escreva uma função que gere e imprima todos os números binários com valores decimais de 1 a  $n$ .

### Exemplo:

- Entrada:  $n = 2$ ; Saída: 1, 10
- Entrada:  $n = 5$ ; Saída: 1, 10, 11, 100, 101

**Método ingênuo** Um método simples é executar um loop de 1 a  $n$  e converter os números de decimal para binário dentro do loop.

**Método Eficiente** Usando uma estrutura de dados da fila:

1. Crie uma fila vazia de strings
2. Enfileirar o primeiro número binário "1" na fila.
3. Agora execute um loop para gerar e imprimir  $n$  números binários.
  - Desenfileire e imprima.
  - Crie uma cópia do item da frente
  - Acrescente "0" no final do item original e coloque-o na fila.
  - Anexar "1" no final do item copiado e enfileirar.

Fonte: [www.geeksforgeeks.org](http://www.geeksforgeeks.org)

**Questão 6.****Questão beecrowd 1110**

Dada uma pilha de  $n$  cartas enumeradas de 1 até  $n$  com a carta 1 no topo e a carta  $n$  na base.

A seguinte operação é realizada enquanto tiver 2 ou mais cartas na pilha.

Jogue fora a carta do topo e mova a próxima carta (a que ficou no topo) para a base da pilha.

Sua tarefa é encontrar a sequência de cartas descartadas e a última carta remanescente.

Cada linha de entrada (com exceção da última) contém um número  $n \leq 50$ . A última linha contém 0 e não deve ser processada. Cada número de entrada produz duas linhas de saída. A primeira linha apresenta a sequência de cartas descartadas e a segunda linha apresenta a carta remanescente.

A entrada consiste em um número indeterminado de linhas contendo cada uma um valor de 1 até 50. A última linha contém o valor 0.

Para cada caso de teste, imprima duas linhas. A primeira linha apresenta a sequência de cartas descartadas, cada uma delas separadas por uma vírgula e um espaço. A segunda linha apresenta o número da carta que restou. Nenhuma linha tem espaços extras no início ou no final. Veja exemplo para conferir o formato esperado.

**Entrada** A entrada será um valor inteiro positivo.

**Saída** A saída será uma sequência de seis números ímpares.

Exemplo de Entrada	Exemplo de Saída
7	Descartadas: 1, 3, 5, 7, 4, 2 Restante: 6
19	Descartadas: 1, 3, 5, 7, 9, 11, 13, 15, 17, 19, 4, 8, 12, 16, 2, 10, 18, 14 Restante: 6
10	Descartadas: 1, 3, 5, 7, 9, 2, 6, 10, 8 Restante: 4
6	Descartadas: 1, 3, 5, 2, 6 Restante: 4
0	