

Fila

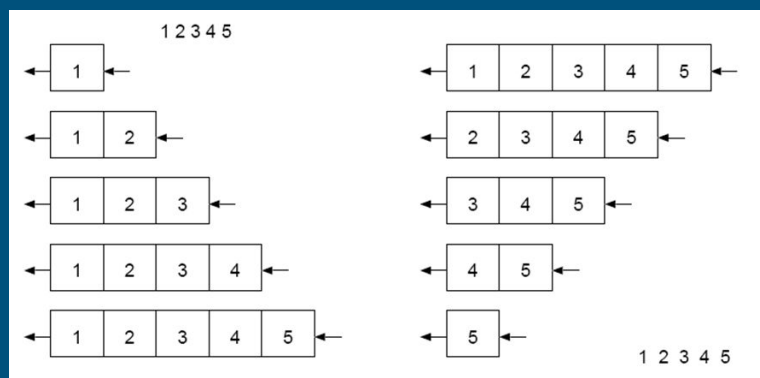
Conjunto ordenado de itens no qual somente em UMA das extremidades novos itens podem ser inseridos, E somente na OUTRA extremidade itens podem ser removidos.

FIFO – First In, First Out - O primeiro a entrar será o primeiro a sair

A extremidade onde os itens são removidos é o começo (início) da fila e a extremidade onde são inseridos é o final da fila.

emerson@paduan.pro.br

Comportamento



emerson@paduan.pro.br

Let's Code!

```
$this->rule_exists( $resource_details['id'], $role_details['id'] ) {
    if ( $access == false ) {
        // Remove the rule as there is currently no need for it
        $details['access'] = false;
        $this->sql->delete( 'acl_rules', $details );
    } else {
        // Update the rule with the new access value
        $this->sql->update( 'acl_rules', array( 'access' => $access ) );
    }
    foreach( $this->rules as $key=>$rule ) {
        if ( $details['role_id'] == $rule['role_id'] && $details['resource_id'] == $rule['resource_id'] ) {
            if ( $access == false ) {
                unset( $this->rules[ $key ] );
            } else {
                $this->rules[ $key ]['access'] = $access;
            }
        }
    }
}
```

emerson@paduan.pro.br

Exemplo

```
public static void main(String[] args) {
    Queue<Pessoa> queueA = new LinkedList<>();

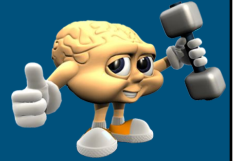
    Random rd = new Random();

    for (int i = 0; i < 5; i++) {
        queueA.add(new Pessoa(rd.nextInt(10), "Pessoa " + i ));
    }

    System.out.println(queueA);
}
```

emerson@paduan.pro.br

Exercício



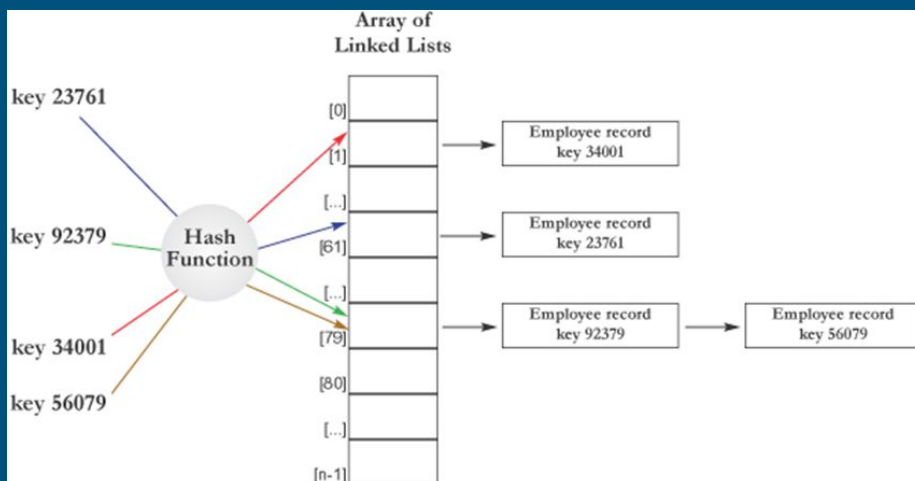
Crie uma classe `FilaDePrioridade` para armazenar objetos da classe `Pessoa` (que contém nome e senha número de atendimento), simulando a chegada e saída de pessoas em uma fila de banco. A classe deve utilizar internamente 2 filas (uma normal e uma de prioridade) utilizando a classe `LinkedList` (Queue) apresentada nesta aula como base.

Ao realizar a remoção da fila, deve ser seguida a seguinte estratégia: devem ser atendidas 2 com prioridade para cada 1 pessoal normal. Caso a fila de prioridades esteja vazia, remova pessoas normais.

Para testar, construa um programa que utilize essa classe `FilaDePrioridade`, fornecendo opções para a entrada de pessoas do tipo normal e prioritária, atender o próximo da fila, e exibir a fila.

emerson@paduan.pro.br

Hash



emerson@paduan.pro.br

Em Java

```
public static void main(String[] args) {  
    HashMap<Integer, String> mapa = new HashMap<>();  
  
    mapa.put(1, "um");  
    mapa.put(2, "dois");  
    mapa.put(3, "três");  
    mapa.put(4, "quatro");  
  
    System.out.println(" 3 = " + mapa.get(3));  
    System.out.println(" 3 = " + mapa.get(32));  
}
```

emerson@paduan.pro.br

Árvores



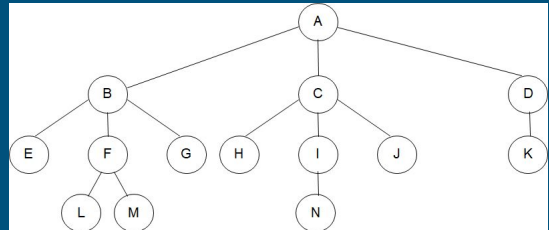
E as árvores ...

emerson@paduan.pro.br

Árvores

Árvores são estruturas de dados formadas por um conjunto finito de elementos denominados nós.

Temos um nó especial chamado raiz da árvore, e os elementos restantes são particionados em conjuntos distintos não vazios, as subárvores, sendo cada um destes conjuntos por sua vez uma árvore.



emerson@paduan.pro.br

Em Java

```
public static void main(String[] args) {  
    int[] nums = {2,4,1,6,3,7,9,5};  
    TreeSet<Integer> tree = new TreeSet<>();  
    //Insere todos os elementos na árvore  
    for (int num : nums) {  
        tree.add(num);  
    }  
    //Exibe todos os elementos da árvore  
    for(int s: tree){  
        System.out.println(s);  
    }  
}
```

emerson@paduan.pro.br