

Programação Orientada a Objetos

LISTA – FILA – PILHA

Prof. M.Sc. Marcos M. Chaves, Esp.GPJ

Resumo

1 Lista → [Fila | Pilha]

Vetor

Recordando

esquemáticamente, um **vetor** é uma área **limitada** de memória (RAM) onde são armazenados conteúdos de **tipos iguais**. O número **5**, no código Java, limita o tamanho do vetor **v**.

	0	1	2	3	4
Vetor v[5]	2	3	5	7	11

Em Java


```
int[] v = new int[5];
```

1.1 Lista (ListaLigada)

Definição

Uma **lista** é uma sequência de elementos (**nós**) organizados de uma determinada maneira. Lembra um **vetor**, porém com a vantagem de não ser necessário limitar o tamanho de elementos. O limite da lista é o tamanho de memória **disponível** para seu crescimento.

Nó

 ou **node**, é o elemento onde um certo conteúdo será armazenado. Nele se armazenam **dados** e um **indicador** (ponteiro) para o **fim** da lista ou para o **próximo** nó da lista.

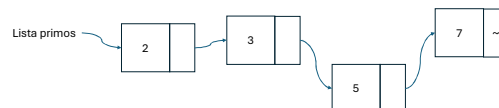
Lista Vazia

é a lista, recém-criada, onde ainda não foram inseridos **nós**.

Lista com um nó



Lista

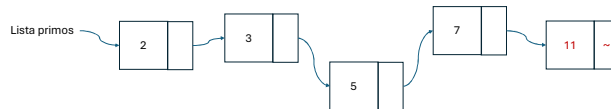


Operações em Listas

em uma **lista** pode-se realizar as seguintes **operações**:

inserir (posição)

onde **posição**=**[início|fim|índice]**



inserir **11** no **fim**

excluir (posição) onde posição=[início|fim|índice]

percorrer () do início ao fim

Instruções Java

```
LinkedList<Classe> objeto = new LinkedList<Classe>();
objeto.add(item)            adiciona item ao final
objeto.addFirst(item)      adiciona item no início
objeto.addLast(item)       adiciona item ao final
objeto.removeFirst()       exclui item do início
objeto.removeLast()        exclui item do final
objeto.getFirst()          retorna o primeiro item da lista
objeto.getLast()            retorna o último item da lista
```

Em Java cria a lista **primos**, insere os números primos sempre ao **final** e **mostra** a lista

```
LinkedList<Integer> primos = new LinkedList<Integer>();

primos.add(2);
primos.add(3);
primos.add(5);
primos.add(7);
primos.add(11);

System.out.println(primos);
```

[2, 3, 5, 7, 11]

Em Java percorre a lista **primos**, mostrando **individualmente** cada número primo armazenado.

```
Iterator<Integer> q = primos.iterator();
while (q.hasNext()) {
    System.out.println(q.next().intValue());
}
```

2
3
5
7
11

1.2 Fila

Definição

Fila é uma implementação de **lista** que segue a estratégia **FIFO**.

Estratégia

First.In.First.Out (FIFO) – o primeiro elemento que entra é o primeiro que sai.

Exemplo

Fila de clientes para o atendimento em uma agência bancária. O **próximo** cliente a ser atendido é a **primeira** pessoa à direita na fila. O cliente **entra** na fila pelo seu **fim**.

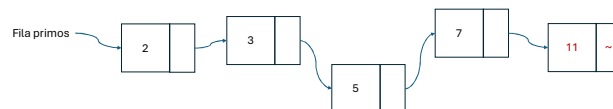


Operações em Filas

em uma **fila** pode-se realizar as seguintes **operações**:

inserir (posição)

onde **posição=fim**



inserir 11 no fim

excluir (posição)

onde **posição=início**

percorrer ()

do **início** ao **fim**

Instruções Java

```
LinkedList<Classe> objeto = new LinkedList<Classe>();  
objeto.add(item)           adiciona item ao final  
objeto.addLast(item)       adiciona item ao final  
objeto.removeFirst()       exclui item do início  
objeto.getFirst()          retorna o primeiro item da fila
```

1.3 Pilha

Definição

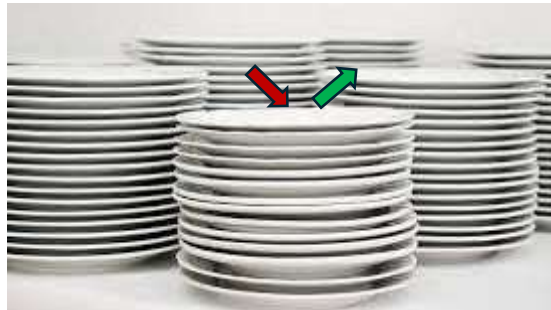
Pilha é uma implementação de **lista** que segue a estratégia **LIFO**.

Estratégia

Last.In.First.Out (LIFO) – o último elemento que entra é o primeiro que sai.

Exemplo

Pilha de pratos lavados e aguardando a secagem e guarda. O próximo prato a ser seco e guardado é o prato do **topo** da pilha. Coloca-se **mais** pratos pelo **topo** da pilha.

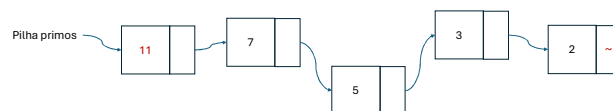


Operações em Pilhas

em uma **pilha** pode-se realizar as seguintes **operações**:

inserir (posição)

onde **posição=topo**



inserir 11 no topo

excluir (posição)

onde **posição=topo**

percorrer ()

do **início** ao **fim**

Instruções Java

```
LinkedList<Classe> objeto = new LinkedList<Classe>();  
objeto.addFirst(item)      adiciona item no topo  
objeto.removeFirst()      exclui item do topo  
objeto.getFirst()         retorna o primeiro item do topo da pilha
```