

1. Considere, a fila dinâmica abaixo. Tendo como restrição a possibilidade de um único tipo abstrato de dados (Estrutura de dados) auxiliar, propor e demonstrar uma solução lógica que inverta os elementos que estão na fila.



2. Simular o comportamento de filas dinâmicas para os algoritmos abaixo (A simulação deve deixar evidente a fila que sobrou na memória):

a)

```
Para (i = 0 ; i < 10 ; i++) {  
    Se (i % 2 == 0) {  
        Insert(i * i);  
    } Senão {  
        Se (i <= 5) {  
            Insert(i);  
        } Senão {  
            Escreva(Remove());  
        }  
    }  
    List();  
}  
Escreva(Size());
```

b)

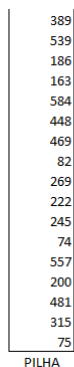
```
Para (i = 100 ; i < 115 ; i++) {  
    Se (isEmpty()) {  
        Insert(i + 100);  
    } Senão {  
        Se (Size() <= 4) {  
            Insert(i + 50);  
        } Senão {  
            Escreva(Remove());  
        }  
    }  
    List();  
}  
Escreva(Size());
```

3. Em Java:

Adaptar o modelo de Fila Dinâmica desenvolvido em aula, com os métodos esperados, para uma Fila Genérica.

Transformar o projeto em uma biblioteca, gerando o JAR Fila.

4. Considere a pilha abaixo, determine uma sequência lógica, e demonstre em teste de mesa, que coloque os elementos da pilha em filas separando por pares e ímpares.



5. Considerando a biblioteca Fila, já criada, faça:

Criar um projeto Java (ChamadoSim) e importe a biblioteca. Esse novo projeto simulará uma necessidade de muitas empresas e instituições que atendem clientes, que é criar um canal de geração de senhas para pessoas com alguma prioridade e pessoas que não tem nenhuma prioridade.

A classe Principal, no package view, deve ter na Main, a criação de 2 filas, a fila e a filaPrioritarios, a inicialização das senhas dos prioritários e dos não prioritários e, por fim, deve dar ao usuário a possibilidade de inserir uma nova senha na fila, uma nova senha na fila de prioritários ou chamar uma pessoa para o atendimento. Um menu deve ser criado.

A classe FilaController deve ter os métodos de validação das operações oferecidas na Main da Classe Principal. Todos os métodos devem receber a fila criada no método Main como parâmetro.

- O método de inserir um novo elemento na fila;
- O método de remover o primeiro elemento da fila, que será atendido. A fila não pode estar vazia;
- O método de chamado que deve seguir a seguinte regra:
 - Fazer 3 chamadas prioritárias para 1 da fila não prioritária
 - Se não houver prioritários, já se deve chamar dos não prioritários

Cliente
+Numero : int
+Nome : String