

Nome: Beatriz Carlos Da Silva.
Estrutura de Dados Tarde

2. Cite duas desvantagens da implementação com alocação sequencial estática resolvidas com a alocação dinâmica.

O que a alocação dinâmica difere da estática:

- Na alocação dinâmica, a alocação de memória pode ser determinada o quanto será necessário durante a execução do programa, economiza memória.
- A quantidade de memória pode ser expandida quando você precisar quando o programa estiver em execução.

3. Indique em que situações é melhor utilizar uma lista encadeada ao invés de uma lista usando vetor.

Situações que precisamos usar listas encadeadas para se tornar computacionalmente eficiente:

- Adicionar um elemento na primeira posição de um vetor, por exemplo, consome muito tempo, pois temos de deslocar todos os outros elementos uma posição para a frente. E isso diminui a performance ao decorrer que a lista se torna extensa.
- Remover um elemento da primeira posição implica em deslocar todos os outros elementos que estão na sua frente para trás.

4. Descreva a saída resultante da seguinte série de operações de pilha: push(5), push(3), pop(), push(2), push(8), pop(), pop(), push(9), push(1), pop(), push(7), push(6), pop(), pop(), push(4), pop(), pop().

```
dom, 22 de mar 20:08
eclipse-workspace - EstruturaDeDadosT/src/Questao1/StackClass.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

terminated TestStack [Java Application] /usr/lib/jvm/java-11-openjdk-amd64/bin/java (22 de mar de 2020 20:07:39)
Atenção sua pilha está vazia!

Aqui está a pilha criada: 5
Aqui está a pilha criada: 3 5
O valor removido foi 3.

Aqui está a pilha criada: 5
Aqui está a pilha criada: 2 5
Aqui está a pilha criada: 8 2 5
O valor removido foi 8.

Aqui está a pilha criada: 2 5
O valor removido foi 2.

Aqui está a pilha criada: 5
Aqui está a pilha criada: 9 5
Aqui está a pilha criada: 1 9 5
O valor removido foi 1.

Aqui está a pilha criada: 9 5
Aqui está a pilha criada: 7 9 5
Aqui está a pilha criada: 6 7 9 5
O valor removido foi 6.

Aqui está a pilha criada: 6 7
Aqui está a pilha criada: 4 6 7
O valor removido foi 4.

Aqui está a pilha criada: 6
O valor removido foi 6.
```

The screenshot shows the Eclipse IDE interface with the console window open. The console output displays a series of operations on a stack, where each element is associated with a creation index. The operations and their results are as follows:

Operation	Creation Index	Value
Aqui está a pilha criada:	5	
Aqui está a pilha criada:	9	5
Aqui está a pilha criada:	1	9 5
O valor removido foi		1.
Aqui está a pilha criada:	9	5
Aqui está a pilha criada:	7	9 5
Aqui está a pilha criada:	6	7 9 5
O valor removido foi		6.
Aqui está a pilha criada:	7	9 5
O valor removido foi		7.
Aqui está a pilha criada:	9	5
Aqui está a pilha criada:	4	9 5
O valor removido foi		4.
Aqui está a pilha criada:	9	5
O valor removido foi		9.
Aqui está a pilha criada:	5	
Aqui está a pilha criada:	5	

5. Descreva a saída resultante da seguinte série de operações de fila: enqueue(5), enqueue(3), dequeue(), enqueue(2), enqueue(8), dequeue(), dequeue(), enqueue(9), enqueue(1), dequeue(), enqueue(7), enqueue(6), dequeue(), dequeue(), enqueue(4), dequeue(), dequeue().

The screenshot shows a console window with the following output:

```
terminated: TestStack (Java Application) /usr/lib/jvm/java-11-openjdk-amd64/bin/java (22 de mar de 2020 20:07:39)
Fila vazia.
5
```