

UTFPR - Estrutura de Dados I

Prof^a Renata Luiza Stange Carneiro Gomes

Atividade 02:

Listas Lineares: Pilha e Fila.

1 Instruções

Esta atividade é individual. Os exercícios devem ser implementados utilizando a linguagem de programação **Java**. Comente as principais linhas do código, explicando o que está sendo implementado. Você deve entregar os arquivos `.java`, sem qualquer estrutura de pacote (package), inclusive no código. A atividade teórica deve estar em um arquivo `pdf`.

2 Atividade: Teórica

2.1 Responda as questões a seguir:

1. O que significa alocação sequencial de memória para um conjunto de elementos?
2. O que significa alocação estática de memória para um conjunto de elementos?
3. Qual a diferença entre alocação sequencial e alocação encadeada?
4. Quais as vantagens de se utilizar alocação encadeada para um conjunto de elementos? Quais as possíveis desvantagens?
5. Qual a diferença entre alocação estática e alocação dinâmica?

2.2 Dada uma pilha de inteiros vazia, o que estará na pilha após as seguintes operações?

- $\text{push}(9) \rightarrow \text{push}(10) \rightarrow \text{pop}() \rightarrow \text{push}(\text{pop()}*2) \rightarrow \text{peek}() \rightarrow \text{push}(30) \rightarrow \text{push}(\text{pop()}/3)$

Desenhe um diagrama da aparência de uma pilha após cada operação:

2.3 Dada uma pilha de inteiros vazia, o que estará na pilha após as seguintes operações?

- $\text{push}(1) \rightarrow \text{push}(\text{pop()}+3) \rightarrow \text{peek}() \rightarrow \text{push}(\text{pop()}/2) \rightarrow \text{pop}() \rightarrow \text{pop}()$

Desenhe um diagrama da aparência de uma pilha após cada operação:

2.4 Dada uma pilha s , S é $s.push(s.pop() + s.pop())$ e T é $s.push(s.pop() * s.pop())$. Qual será o resultado após as seguintes operações?

- $s.push(1) \rightarrow s.push(2) \rightarrow s.push(3) \rightarrow s.push(4) \rightarrow s.push(5) \rightarrow S \rightarrow T \rightarrow S \rightarrow T$

2.5 A classe *Stack* da biblioteca Java implementa o tipo abstrato de pilha e as operações *push* e *pop*. Considere o exemplo abaixo.

```
import java.util.Stack;

public class Pilha {

    public static void main(String args[]) {

        Stack<String> s = new Stack<String>();
        s.push("Ana");
        s.push("Paulo");
        s.push("Pedro");
        s.push("Marcos");

        while (s.size()>0){
            System.out.println(s.pop());
        }

    }
}
```

2.5.1 Sobre a classe *Pilha* apresentada é correto afirmar:

- () instrução *s.pop(1)*; poderia ter sido utilizada antes do comando *while* para excluir o nome Paulo da pilha *s*.
- () Ao término da execução do programa a pilha *s* estará vazia.
- () Ocorrerá um erro em tempo de execução, pois o método *pop* apenas remove o último elemento que entrou na pilha e não retorna valor a ser exibido.
- () Ao término da execução do programa a pilha *s* permanecerá com os quatro nomes, pois o método *pop* apenas retorna o elemento da pilha, sem removê-lo.

Justifique suas respostas!

2.6 Dada uma pilha com 15 elementos e uma fila com 30 elementos, execute o seguinte conjunto de operações 60 vezes:

- *enqueue(pop())* /*remove o elemento do topo da pilha e o insere na fila
- *push(dequeue())* /*remove o último elemento da fila e o insere na pilha

2.6.1 Quantos dos 45 elementos totais estão envolvidos neste processo?

Observe que NÃO está perguntando *quantas vezes* os elementos se movimentam, mas *quantos elementos* se movimentam.