**Atividades iguais serão zeradas.**

**Exercícios que envolvam codificações iguais serão zerados.**

1) Simule o método a seguir, demonstrando cada passo em uma representação gráfica de lista:

public void metodoA (int pos, int valor){

if (dados.length == tamanho)

System.out.println("ERRO!”);

else {

if (pos <= 0 || pos > tamanho+1)

System.out.println("Posição Inválida!");

else

if (pos = = 1){

adicionaInicio(valor);

else

if (pos = = tamanho)

adicionaFinal(valor);

else{

for (int i=tamanho; i>=pos;i--)

dados[i]=dados[i-1];

dados[pos-1]=valor;

tamanho++;

}

}

}

}

2) Implemente uma Lista de caracteres em alocação estática com todas as operações indicadas a seguir:

-verificar se a lista está vazia, retornando true se estiver vazia e false se não estiver;

-verificar se a lista está cheia, retornando true se estiver cheia e false se não estiver;

-adicionar caractere no início da lista, caso a operação não possa ser realizada, mostre mensagem avisando;

-adicionar caractere no final da lista, caso a operação não possa ser realizada, mostre mensagem avisando;

-adicionar caractere em determinada posição da lista, caso a operação não possa ser realizada, mostre mensagem avisando o motivo;

-remover caractere do início da lista, retornando o elemento que foi removido, caso a operação não possa ser realizada, mostre mensagem avisando;

-remover caractere do final da lista, retornando o elemento que foi removido, caso a operação não possa ser realizada, mostre mensagem avisando;

-remover caractere de determinada posição da lista, retornando o elemento que foi removido, caso a operação não possa ser realizada, mostre mensagem avisando;

-percorrer a lista concatenando os elementos em uma String que será devolvida no final.

3) Implemente uma Lista de Strings em alocação estática com todas as operações indicadas a seguir:

-verificar se a lista está vazia, retornando true se estiver vazia e false se não estiver;

-verificar se a lista está cheia, retornando true se estiver cheia e false se não estiver;

-adicionar uma String no início da lista, caso a operação não possa ser realizada, mostre mensagem avisando;

-adicionar uma String no final da lista, caso a operação não possa ser realizada, mostre mensagem avisando;

-adicionar uma String em determinada posição da lista, caso a operação não possa ser realizada, mostre mensagem avisando o motivo;

-remover a String do início da lista, retornando o elemento que foi removido, caso a operação não possa ser realizada, mostre mensagem avisando;

-remover a String do final da lista, retornando o elemento que foi removido, caso a operação não possa ser realizada, mostre mensagem avisando;

-remover a String de determinada posição da lista, retornando o elemento que foi removido, caso a operação não possa ser realizada, mostre mensagem avisando;

-percorrer a lista concatenando os elementos em uma String que será devolvida no final.