**INSTITUTO SUPERIOR DE TRANSPORTES E COMUNICAÇÕES

**Resolução da Ficha resumo 2**

AUTOR**: SHELTON UAMUSSE LECC-11**

**String (Incompleto)**

1. Fazer uma classe que:

* Receba duas string do usuário;
* Conta e informa quantas vezes a segunda string ocorre dentro da primeira;
* Informa uma estatística dos caracteres contidos nas duas strings.

1. Faça um programa que receba um nome completo na forma de uma String e mostre a abreviatura deste nome. Não se devem abreviar as palavras com 2 ou menos letras. A abreviatura deve vir separada por pontos. Ex Paulo José de Almeida Prado. Abreviatura: P.J. de A.P.
2. Fazer um programa que receba uma String do usuário e mostre o conteúdo desta String de forma invertida.

package string;

import java.util.Scanner;

public class ex3Inverter {

public static void main(String[] args) {

System.***out***.println("Programa para inverter!");

System.***out***.println("");

System.***out***.println("Insira a palavra que deseja receber o seu inverso: ");

String palavra=new Scanner(System.***in***).next();

String palavraInversa=" ";

System.***out***.println("");

System.***out***.println("Palavra inserida: "+palavra);

System.***out***.println("");

String inveritida=*inverterPalavra*(palavra,palavraInversa);

System.***out***.println("Palavra invertida: "+inveritida);

}

public static String inverterPalavra (String palavra,String palavraInversa) {

for(int i=palavra.length()-1;i>=0;i--) {

palavraInversa=palavraInversa+palavra.charAt(i);

}

return (palavraInversa);

}

}

1. Um dos sistemas de encriptação mais antigos é atribuído a Júlio César: se uma letra a ser encriptada é a letra de número N do alfabeto, substitua- a com a letra (N+K), onde k é um número inteiro constante (César utilizava K=3). Usualmente consideramos o espaço como zero e todos os cálculos são realizados com módulo -27. Dessa forma, para k=1 a mensagem “Ataque ao amanhecer” se torna “bubrfabpabnboifdfs”. Faça um programa que receba como entrada uma mensagem de valor j e retorne a mensagem criptografada pelo código de César. Fraquezas: apenas 26 chaves possíveis. É possível utilizar conhecimento da linguagem para facilitar a busca.
2. Faça um programa que solicite o nome do usuário e em seguida mostrar o nome em formato de escada:

F

FU

FUL

FULA

FULAN

FULANO

import java.util.Scanner;

public class Ex7S {

public static void main(String[] args) {

System.***out***.println("Insira uma palavra: ");

String word=new Scanner(System.***in***).next();

for(int i=0;i<word.length();i++) {

System.***out***.println(word.substring(0,i+1));

}

}

}

1. Altere o programa anterior de modo que a escada seja invertida:

FULANO

FULAN

FULA

FUL

FU

F

import java.util.Scanner;

public class Ex6S {

public static void main(String[] args) {

System.out.println("Insira uma palavra: ");

String word=new Scanner(System.in).next();

for(int i=word.length();i>0;i--) {

System.out.println(word.substring(0,i));

}

}

}

**Array unidimensional**

1. Crie um método que recebe um array de inteiros e retorna a quantidade de elementos do array que são números negativos.

import java.util.Scanner;

public class ex1AB{

public static Scanner ler=new Scanner(System.in);

public static int contNegativo=0, somaP=0;

public static void main(String [] args){

System.out.println("Deseja inserir quantos valores : ");

int n=new Scanner(System.in).nextInt();

int V[]=new int [n];

System.out.println("Insira "+n+" nrs: ");

for(int i=0;i<V.length;i++){

V[i]=ler.nextInt();

if(V[i]<0){

contNegativo=contNegativo+1;

}else{

somaP=somaP+V[i];

}

}

System.out.println("O conjunto tem "+contNegativo+" nrs negativos");

System.out.println("A soma dos positivos e "+somaP);

}

}

1. Para cada conjunto de valores, escreva um código java, que preencha um array de forma automática com os valores:
2. Crie um método que recebe um array A de inteiros e um valor inteiro x e retorna a quantidade de vezes que x aparece no array A.
3. Escreva um método que recebe um array A de inteiros e devolve um array de boolean onde, cada posição indique true se o elemento da posição correspondente de A é positivo e false caso seja negativo ou zero.

import java.util.\*;

public class ex6{

public static Scanner ler=new Scanner(System.in);

public static void main(String[] args){

System.out.println("Quantos nrs deseja inserir: ");

int n=ler.nextInt();

int v[]=new int[n];

boolean x[]=new boolean[n];

for(int i=0;i<n;i++){

System.out.println("Insira o"+(i+1)+"o nr");

v[i]=ler.nextInt();

}

preencherX(n,v,x);

}

public static void preencherX(int n,int v[],boolean x[]){

for(int i=0;i<n;i++){

if(v[i]<=0){

x[i]=false;

}else{

x[i]=true;

}

}

System.out.println("O vector de inteiros: "+Arrays.toString(v));

System.out.println("O vector de booleans: "+Arrays.toString(x));

}

}

1. Escreva um método que recebe um array de números e devolve a posição onde se encontra o maior valor do array. Se houver mais de um valor maior, devolver a posição da primeira ocorrência.

import java.util.\*;

public class ex5{

public static int maior=0,maior1=0,posicao=0;

public static Scanner ler=new Scanner(System.in);

public static void main(String[] args){

System.out.println("Insira a quantidade de elementos que deseja: ");

int n=ler.nextInt();

int v[]=new int[n];

preencher(v);

mostrar(v);

procurarMaior(v);

}

public static void preencher(int v[]){

System.out.println("Preencha o vector: ");

for(int i=0;i<v.length;i++){

System.out.println("Insira o "+(i+1)+"o nr");

v[i]=ler.nextInt();

}

}

public static void mostrar(int v[]){

System.out.println("O vector tem os seguintes nrs: "+Arrays.toString(v));

}

public static void procurarMaior(int v[]){

maior=v[0];

for(int i=1;i<v.length;i++){

if(v[0]<v[i]){

maior=v[i];

posicao=posicao+1;

}

}

System.out.print("O maior nr ="+maior+" e esta na posicao "+posicao);

}

}

1. Crie um método que recebe um array de inteiros positivos e substitui seus elementos de valor ímpar por -1 e os pares por +1.

import java.util.Scanner;

import java.util.Arrays;

public class ex4{

public static Scanner ler=new Scanner(System.in);

public static int n;

public static void main(String[] args){

System.out.println("Insira a quantidade de nrs que pretende inserir");

int n=ler.nextInt();

int V[]=new int[n];

int Trocar[]=new int[n];

preencherV(n,V);

substituirV(n,V,Trocar);

mostrarVeT(n,V,Trocar);

}

public static void preencherV(int n, int V[]){

System.out.println("Insira "+n+" nrs: ");

for(int i=0;i<n;i++){

V[i]=ler.nextInt();

}

}

public static void substituirV(int n, int V[], int Trocar[]){

for(int i=0;i<n;i++){

if(V[i]%2!=0){

Trocar[i]=-1;

}else{

Trocar[i]=1;

}

}

}

public static void mostrarVeT(int n,int V[], int Trocar[]){

System.out.println("Antes da trocar o vector era: "+Arrays.toString(V));

System.out.println("Depois da trocar o vector ficou: "+Arrays.toString(Trocar));

}

}

1. Considere que tem uma tabela de 10 inteiros que é inicializada com alores aleatórios. Escreva um programa para ordenar esta tabela.
2. Faça um programa de consulta pela posição numérica da pessoa: leia os nomes, sendo a quantidade determinada pelo usuário. Logo após a entrada pergunte ao usuário o número da pessoa que ele gostaria de consultar. Após a sua resposta, exiba o nome que fica na posição informada. Chame atenção do usuário em caso de uma consulta inválida, ou seja, com números menores ou iguais a zero, ou maiores que a quantidade cadastrada.
3. Faça um programa de consulta de telefone a partir de um nome informado por uma chave de dados: leia nomes das pessoas com os seus respectivos números, sendo a quantidade determinada pelo usuário. Em seguida pergunte ao usuário qual o nome que ele deseja consultar o telefone. Após a sua resposta, exiba o telefone da pessoa procurada.

**Array Bidimensional**

1. Escreva um programa em java que leia uma matriz de inteiros. O programa deve indicar se a matriz é uma identidade ou não.

import java.util.Scanner;

public class EX1AB {

public static Scanner ler = new Scanner(System.in);

public static void main(String[] args) {

System.out.println("Defina o tamanho da matriz: ");

int m=ler.nextInt();

int n=ler.nextInt();

int matriz[][]=new int[m][n];

preencherEMostrarMatriz(m,n,matriz);

}

public static void preencherEMostrarMatriz(int m,int n,int matriz[][]) {

System.out.println("Preencha a matriz: ");

for(int i=0;i<m;i++) {

System.out.println("Preencha a linha "+(i+1));

for(int j=0;j<n;j++) {

matriz[i][j]=ler.nextInt();

}

}

testeDaMatriz(m,n,matriz);

}

public static void testeDaMatriz(int m,int n,int matriz[][]) {

boolean det1=false;

boolean det2=false;

for(int i=0;i<m;i++) {

for(int j=0;j<n;j++) {

if(i==j && matriz[i][j]==1 && i!=j && matriz[i][j]==0) {

det1=true;

}

}

}

if(det1==true) {

System.out.println("A matriz é identidade");

}else {

System.out.println("A matriz não é identidade");

}

}

}

1. Escreva um programa em java que leia duas matrizes de inteiros. O programa deve indicar se as duas matrizes são iguais ou diferentes. Se forem diferentes deve indicar os valores diferentes.
2. Declarar e construir uma matriz de 10 inteiros, solicitar o usuário que informe os valores dos elementos e mostrar os índices do maior e do menos.

import java.util.Scanner;

public class EX3AB {

public static Scanner *ler*=new Scanner(System.***in***);

public static void main(String[] args) {

int matriz[][]=new int[10][10];

*preencherMatriz*(matriz);

}

public static void preencherMatriz(int matriz[][]) {

System.***out***.println("Preencha as 10 linhas e 10 colunas");

for(int i=0;i<10;i++) {

for(int j=0;j<10;j++) {

matriz[i][j]=*ler*.nextInt();

}

}

System.***out***.println("A matriz preenchida: ");

for(int i=0;i<10;i++) {

for(int j=0;j<10;j++) {

System.***out***.print(matriz[i][j]+" ");

}

System.***out***.println("");

}

*MaiorMenor*(matriz);

}

public static void MaiorMenor(int matriz[][]) {

int maior=0;

int menor=0;

for(int i=0;i<10;i++) {

for(int j=0;j<10;j++) {

if(matriz[i][j]>maior) {

maior++;

}else {

menor++;

}

}

}

System.***out***.println("Indice do maior: "+maior);

System.***out***.println("Indice do menor: "+menor);

}

}

1. Escreva um programa que transfere os valores de uma Array Bidimensional de duas dimensões para um Array Unidimensional.

import java.util.\*;

public class Ex4PassarParaAU {

public static void main(String[] args) {

//Scanner ler=new Scanner(System.in);

System.***out***.println("Defina o tamanho do conjunto, indicando as linhas e colunas: ");

Scanner scanner = new Scanner(System.***in***);

int m=scanner.nextInt();

int n=scanner.nextInt();

System.***out***.println("Os valores serão preenchidos automaticamente: ");

int v1[][]=new int[m][n];

int v2[]=new int[m\*n];

*preencherImprimirv1*(v1,m,n);

*substituirv1ParaV2*(v1,v2,m,n);

}

public static void preencherImprimirv1(int v1[][],int m, int n) {

for(int i=0;i<m;i++) {

for(int j=0;j<n;j++) {

v1[i][j]=j+1;

}

}

System.***out***.println("Os valores na tabela: ");

for(int i=0;i<m;i++) {

for(int j=0;j<n;j++) {

System.***out***.print(v1[i][j]+" ");

}

System.***out***.println("");

}

}

public static void substituirv1ParaV2(int v1[][],int v2[],int m, int n) {

int k=0;

for(int i=0;i<m;i++) {

for(int j=0;j<n;j++) {

v2[k++]=v1[i][j];

}

}

System.***out***.println("Os valores do vector "+Arrays.*toString*(v2));

}

}

**ArrayList- Resolução de exercícios.**

1. Escreva um programa em java que crie um ArrayList, adicione algumas cores e as imprima na tela. Adicione um método para ordenar o ArrayList de cores pelo nome.

import java.util.\*;

public class arrayList1{

public static void main(String[] args){

Scanner ler=new Scanner(System.in);

ArrayList<String>nomes=new ArrayList<String>();

System.out.println("Quantas cores deseja inserir: ");

int n=ler.nextInt();

System.out.println("Insira "+n+" cores: ");

for(int i=0;i<n;i++){

String cor=ler.next();

if(!nomes.contains(cor)){

nomes.add(cor);

}

}

Collections.sort(nomes);

System.out.println("As cores inseridas pelo usuario sao: "+nomes);

}

}

1. Escreva um programa que recupere o maior e menor valor de um ArrayList de inteiros exibindo na tela.

import java.util.\*;

public class ex2{

public static Scanner ler=new Scanner(System.in);

public static void main(String[] args){

System.out.println("Insira a quantidade de nrs que pretende inserir: ");

int n=ler.nextInt();

ArrayList<Integer>nrs=new ArrayList<>();

for(int i=0;i<n;i++){

System.out.println("Insira o "+(i+1)+"o nr: ");

nrs.add(ler.nextInt());

}

acharMaior(nrs);

acharMenor(nrs);

}

public static void acharMaior(ArrayList<Integer>nrs){

System.out.print("O maior valor da lista e: ");

int maior=nrs.get(0);

for(int i=1;i<nrs.size();i++){

if(nrs.get(i)>nrs.get(0)){

maior=nrs.get(i);

}

}

System.out.println(maior);

}

public static void acharMenor(ArrayList<Integer>nrs){

System.out.print("O menor valor da lista e: ");

int menor=nrs.get(0);

for(int i=1;i<nrs.size();i++){

if(nrs.get(i)<nrs.get(0)){

menor=nrs.get(i);

}

}

System.out.print(menor);

}

}

1. Escreva um programa que calcule a média dos elementos de um ArrayList.

import java.util.\*;

public class ex3{

public static Scanner ler=new Scanner(System.in);

public static void main(String[] args){

System.out.println("Quantos nrs pretende inserir: ");

int n=ler.nextInt();

ArrayList<Integer>valor;

valor=new ArrayList<>();

for(int i=0;i<n;i++){

System.out.println("Insira o "+(i+1)+"o valor");

valor.add(ler.nextInt());

}

System.out.print("A media dos valores "+valor+" = "+mostrarMedia(valor));

}

public static double mostrarMedia(ArrayList<Integer>valor){

double media=0;

int soma=0;

for(int i=0;i<valor.size();i++){

soma=soma+valor.get(i);

}

media=soma/valor.size();

return media;

}

**Recursividade- Resolução de exercícios**

1. Desenvolva um programa que a soma dos N primeiros números.
2. Desenvolva um programa que dados dois números, x e y, calcule a potência do primeiro número pelo segundo (x^y). Obs: Considere a não existência de uma função exponencial.
3. Desenvolva um programa que calcule o factorial de um número n.
4. Desenvolva um programa que calcule o n-ésimo termo da série de fibonacci.

* Série de Fibonacci: 0, 1, 1, 2, 3, 5, 8, 13, 21, 34...
* Obs: O n-ésimo termo é determinado a partir dos anteriores.

1. Desenvolva um programa que calcule a soma e a multiplicação dos elementos (inteiros) de uma arranjo (array) de tamanho N.
2. Faça um programa (utilizando recursividade) que peça para usuário digitar um número, em seguida, faça a soma de todos os algarismos do número.

Exemplos:

1111 = 1+1+1+1 = 4

2090= 2+0+9+0 = 11

1. Desenvolva um algoritmo que calcule o mostre o maior elemento de um arranjo (array) de tamanho N. Os elementos são números inteiros.