****

**UNIVERSIDADE FEDERAL DE MINAS GERAIS**

Instituto de Ciências Exatas - ICEX

Departamento de Ciência da Computação – DCC

***LISTA 2 – PDROGRAMAÇÃO E DESENVOLVIMENTO DE SOFTWARE 1***

Aluno: Marcone Márcio da Silva Faria

Matrícula: 2019021573

|  |
| --- |
| Exercício 1 |

#include <stdio.h>

#include <stdlib.h>

#include <math.h>

#define N 15

int numerosPares(int[]);

int somaImpar(int[]);

int valorMaximo(int[]);

int maiorValor(int[]);

int maiorDif(int[]);

int main(){

int vetor[N], i,j;

for(i=0; i < N; i++){

vetor[i] = rand()%10;

}

for(j=0; j < N; j++){

printf("%d\t", vetor[j]);

}

printf("\n\na)Quantidade de numeros pares: %d\n", numerosPares(vetor));

printf("b)Soma dos numeros impares: %d\n", somaImpar(vetor));

printf("c)Quantidade de numeros com valor maior que a media: %d\n", valorMaximo(vetor));

printf("d)O maior valor do vetor: %d\n", maiorValor(vetor));

printf("e)A maior diferenca em valor absoluto entre os elementos consecutivos do vetor: %d\n", maiorDif(vetor));

}

int numerosPares(int pares[]){

int k, cont=0;

for(k=0; k < N; k++){

if(pares[k]%2 == 0){

cont++;

}

}

return cont;

}

int somaImpar(int impar[]){

int k, soma=0;

for(k=0; k < N; k++){

if((impar[k])%2 != 0){

soma = soma + impar[k];

}

}

return soma;

}

int valorMaximo(int max[]){

int k,j, cont=0;

float media=0;

for(k=0; k < N; k++){

media = media + max[k];

}

media = media/N;

for(j=0; j < N; j++){

if(max[j] > media){

cont++;

}

}

return cont;

}

int maiorValor(int maior[]){

int k, maiorValor=0;

for(k=0; k < N; k++){

if(maior[k] > maiorValor){

maiorValor = maior[k];

}

}

return maiorValor;

}

int maiorDif(int dif[]){

int k, a=0, b=0, j=0, maiorDif=0, absoluto=0;

for(k=0; k < N; k++){

for(j=k+1; j<N; j++){

absoluto = abs(dif[j] - dif[k]);

if(maiorDif < absoluto){

maiorDif = absoluto;

}

}

}

return maiorDif;

}

|  |
| --- |
| Exercício 1.1 |

#include <stdio.h>

#include <stdlib.h>

#define N 15

int terceiroMaior(int[]);

void ordenarVetor(int[]);

int elementosRepetidos(int[]);

int main(){

int vetor[N], i,j;

for(i=0; i < N; i++){

vetor[i] = rand()%10;

}

for(j=0; j < N; j++){

printf("%d\t", vetor[j]);

}

printf("\na)Exibir o 3o. maior elemento do vetor: %d\n", terceiroMaior(vetor));

printf("b)Ordenar o vetor:\n");

ordenarVetor(vetor);

printf("\nc)Eliminar Repetidos:\n");

printf("%d\t", elementosRepetidos(vetor));

}

int terceiroMaior(int terceiro[]){

int k, maiorValor=0, segundoMaior=0, terceiroMaior=0;

for(k=0; k < N; k++){

if(terceiro[k] > maiorValor){

maiorValor = terceiro[k];

}

if((terceiro[k] != maiorValor) && (terceiro[k] > segundoMaior)){

segundoMaior = terceiro[k];

}

if((terceiro[k] != maiorValor) && (terceiro[k] != segundoMaior) && (terceiro[k] > terceiroMaior)){

terceiroMaior = terceiro[k];

}

}

return terceiroMaior;

}

void ordenarVetor(int ordem[]){

int i, j, aux=0;

for(i=0; i<N; i++){

for(j=i+1; j<N; j++){

if(ordem[i] > ordem[j]){

aux = ordem[i];

ordem[i] = ordem[j];

ordem[j] = aux;

}

}

}

for(i=0; i<N; i++){

printf("%d\t", ordem[i]);

}

}

int elementosRepetidos(int vetor[]){

int i, j, n=0, repete=0, copia[N],x=0;

for(i=0; i<N; i++){

copia[i] = vetor[i];

printf("%d\t", copia[i]);

}

for(i=0; i<N; i++){

for(j=i+1; j<N; j++){

if(vetor[i] == vetor[j]){

copia[i] = 0;

}

}

}

for(i=0; i<N; i++){

printf("%d\t", copia[i]);

}

}

|  |
| --- |
| Exercício 2 |

#include <stdio.h>

#include <stdlib.h>

#define N 3

int main(){

int matriz1[N][N], matriz2[N][N], soma[N][N], i, j, linha=0;

printf("Preencha a matriz 1:\n");

for(i=0; i < N; i++){

for (j=0; j < N; j++){

printf("Digite um valor para a %da linha e para a %da coluna:\n", i+1, j+1);

scanf ("%d", &matriz1[i][j]);

}

}

printf("\n\nPreencha a matriz 2:\n");

for(i=0; i < N; i++){

for (j=0; j < N; j++){

printf("Digite um valor para a %da linha e para a %da coluna:\n", i+1, j+1);

scanf ("%d", &matriz2[i][j]);

}

}

for(i=0; i < N; i++){

for (j=0; j < N; j++){

soma[i][j] = matriz1[i][j] + matriz2[i][j];

}

}

printf("\n\nSOMA DAS MATRIZES 1 E 2:\n");

for(i=0; i < N; i++){

for (j=0; j < N; j++){

if(linha == N){

printf("\n");

linha=0;

}

printf("%d \t", soma[i][j]);

linha++;

}

}

}

|  |
| --- |
| Exercício 3 |

#include <stdio.h>

#include <stdlib.h>

#define N 4

int elementosPares(int[][N]);

int somaImpares(int[][N]);

int maiorQMedia(int[][N]);

int maiorValor(int[][N]);

int segmaiorValor(int[][N]);

int eliminaRep(int[][N]);

int main(){

int matriz[N][N], i, j, menu, linha=0;

for(i=0; i < N; i++){

for (j=0; j < N; j++){

matriz[i][j] = rand()%10;

}

}

for(i=0; i < N; i++){

for (j=0; j < N; j++){

if(linha == N){

printf("\n");

linha=0;

}

printf("%d \t", matriz[i][j]);

linha++;

}

}

printf("\n\nInsira a opcao desejada de 1 a 7:\n");

scanf("%d", &menu);

switch(menu){

case 1:

printf("\n\n1. a quantidade de numeros pares da matriz: %d", elementosPares(matriz));

break;

case 2:

printf("\n2. a soma dos numeros impares da matriz: %d", somaImpares(matriz));

break;

case 3:

printf("\n3. a quantidade de números com valor maior do que a media: %d", maiorQMedia(matriz));

break;

case 4:

printf("\n4. o maior valor da matriz: %d", maiorValor(matriz));

break;

case 5:

printf("\n5. o segundo maior valor da matriz: %d", segmaiorValor(matriz));

break;

case 6:

printf("\n6. eliminar os numeros repetidos da matriz: %d", eliminaRep(matriz));

break;

case 7:

system("pause");

break;

default:

printf("OPCAO INVALIDA!!!");

break;

}

}

int elementosPares(int matriz[][N]){

int k,z,cont=0;

for(k=0; k < N; k++){

for (z=0; z < N; z++){

if(matriz[k][z]%2 == 0){

cont++;

}

}

}

return cont;

}

int somaImpares(int matriz[][N]){

int k,z,soma=0;

for(k=0; k < N; k++){

for (z=0; z < N; z++){

if(matriz[k][z]%2 != 0){

soma = soma + matriz[k][z];

}

}

}

return soma;

}

int maiorQMedia(int matriz[][N]){

int k,z,media=0, soma=0, maior=0, cont=0;

for(k=0; k < N; k++){

for (z=0; z < N; z++){

soma = soma + matriz[k][z];

}

}

media = soma/(N\*N);

for(k=0; k < N; k++){

for (z=0; z < N; z++){

if(matriz[k][z] > media){

cont++;

}

}

}

return cont;

}

int maiorValor(int matriz[][N]){

int k,z,maiorValor=0;

for(k=0; k < N; k++){

for (z=0; z < N; z++){

if(matriz[k][z] > maiorValor){

maiorValor = matriz[k][z];

}

}

}

return maiorValor;

}

int segmaiorValor(int matriz[][N]){

int k,z,maiorValor=0, segmaiorValor=0;

for(k=0; k < N; k++){

for (z=0; z < N; z++){

if(matriz[k][z] > maiorValor){

maiorValor = matriz[k][z];

}

if((matriz[k][z] != maiorValor) && (matriz[k][z] >segmaiorValor)){

segmaiorValor = matriz[k][z];

}

}

}

return segmaiorValor;

}

int eliminaRep(int matriz[][N]){

int i, j, k, z, n=0, repete=0, copia[N][N],x=0, linha=0;

for(i=0; i<N; i++){

for(j=0; j<N; j++){

copia[i][j] = matriz[i][j];

if(linha == N){

printf("\n");

linha=0;

}

printf("%d\t", copia[i][j]);

linha++;

}

}

for(i=0; i<N; i++){

for(j=0; j<N; j++){

for(k=i+1; k<N; k++){

for(z=j+1; z<N; z++){

if(matriz[i][j] == matriz[k][z]){

copia[i][j] = 0;

}

}

}

}

}

for(i=0; i<N; i++){

for(j=0; j<N; j++){

if(linha == N){

printf("\n");

linha=0;

}

printf("%d\t", copia[i][j]);

linha++;

}

}

}

|  |
| --- |
| Exercício 4 |

#include <stdio.h>

#include <stdlib.h>

#define I 12

#define J 4

int main(){

int matriz[I][J], i, j, coluna=0, totalMes=0, cont=0, soma=0, linha=0, totalSemana=0;

for(i=0; i<I; i++){

for (j=0; j < J; j++){

matriz[i][j] = rand()%10;

}

}

for(i=0; i<I; i++){

for (j=0; j<J; j++){

if(coluna < J){

totalMes = totalMes + matriz[i][j];

}

if(coluna == J){

printf("\n");

coluna=0;

printf("Total vendido no mes %d: %d\n", i, totalMes);

totalMes=0;

}

if(coluna==0){

totalSemana = totalSemana + matriz[i][0];

}

soma = soma + matriz[i][j];

printf("%d \t", matriz[i][j]);

coluna++;

}

}

printf("\n\nTotal vendido no ano:%d", soma);

printf("\n\nTotal vendido nas primeiras semanas dos meses: %d\n", totalSemana);

}

|  |
| --- |
| Exercício 5 |

#include <stdio.h>

#include <stdlib.h>

#define I 8

#define J 12

int main(){

int aux=0, linha=0, i, j, matriz[I][J], mes=0, ano=0;

float consumoMedio=0, vconsumoMedio[J], consumoAno=0, vconsumoAno[I], maiorAno=0, maiorMes=0;

for(i=0; i<I; i++){

for (j=0; j<J; j++){

matriz[i][j] = rand()%10;

}

}

for (j=0; j<J; j++){

for(i=0; i<I; i++){

consumoMedio = consumoMedio + matriz[i][j];

if(aux == J){

printf("\n");

aux = 0;

}

printf("%d\t", matriz[i][j]);

aux++;

}

vconsumoMedio[j] = (consumoMedio/I);

consumoMedio = 0;

}

for (i=0; i<I; i++){

for(j=0; j<J; j++){

consumoAno = consumoAno + matriz[i][j];

}

vconsumoAno[i] = (consumoAno/J);

consumoAno = 0;

}

printf("\n\n");

for(j=0; j<J; j++){

if(vconsumoMedio[j] > maiorMes){

maiorMes = vconsumoMedio[j];

mes = j;

}

printf("Consumo medio do mes %d: %.2f\n", j+1 ,vconsumoMedio[j]);

}

printf("\n\n");

for(i=0; i<I; i++){

if(vconsumoAno[i] > maiorAno){

maiorAno = vconsumoAno[i];

ano = i;

}

printf("Consumo medio do ano %d: %.2f\n", i+1 ,vconsumoAno[i]);

}

if(maiorAno > maiorMes){

printf("\nAno com maior gasto de energia (comparado ao consumo por mes e ano): %d", ano);

}else{

if(maiorAno == maiorMes){

printf("\nAno/Mes com maior gasto de energia (comparado ao consumo por mes e ano): %d", ano);

}else{

printf("\nMes com maior gasto de energia (comparado ao consumo por mes e ano): %d", mes);

}

}

}

|  |
| --- |
| Exercício 6 |

#include <stdio.h>

#include <stdlib.h>

#define M 2

#define N 3

int main(){

int i, j, matriz[M][N], coluna=0, v1[N], v2[N], maiorValor=0, menorValor=0;

int vmaiorValor[N], vmenorValor[N];

for(i=0; i<M; i++){

for (j=0; j<N; j++){

matriz[i][j] = rand()%10;

}

}

for(j=0; j<N; j++){

for (i=0; i<M; i++){

maiorValor = matriz[0][0];

menorValor = matriz[0][0];

if(coluna == N){

printf("\n");

coluna=0;

}

printf("%d \t", matriz[i][j]);

coluna++;

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

maiorValor = matriz[i][j];

}

if(matriz[i][j] < menorValor){

menorValor = matriz[i][j];

}

}

vmaiorValor[j] = maiorValor;

vmenorValor[j] = menorValor;

}

printf("\n\nMenores Valores:\n");

for(j=0; j<N; j++){

printf("%d\t", vmenorValor[j]);

}

printf("\nMaiores Valores:\n");

for(j=0; j<N; j++){

printf("%d\t", vmaiorValor[j]);

}

}

|  |
| --- |
| Exercício 7 |

#include <stdio.h>

#include <stdlib.h>

#define I\_1 12

#define J\_1 2

#define I\_2 2

#define J\_2 2

int main(){

int i, j, aux=0, matriz\_1[I\_1][J\_1], matriz\_2[I\_2][J\_2], linha=0, lucro\_M1=0, lucro\_M2=0, tot\_1=0, tot\_2=0;

for(i=0; i<I\_1; i++){

for (j=0; j < J\_1; j++){

matriz\_1[i][j] = rand()%10;

}

}

for(i=0; i<I\_2; i++){

for (j=0; j < J\_2; j++){

matriz\_2[i][j] = rand()%10;

}

}

printf("TABELA 1:\n");

for(i=0; i<I\_1; i++){

for (j=0; j<J\_1; j++){

if(i <= I\_1){

if(j==0){

lucro\_M1 = ((matriz\_1[i][0])\*(matriz\_2[0][1]));

tot\_1 += lucro\_M1;

/\*printf("Lucro M1 = %d\n", lucro\_M1);\*/

}

if(j==1){

lucro\_M2 = ((matriz\_1[i][1])\*(matriz\_2[1][1]));

tot\_2 += lucro\_M2;

/\*printf("Lucro M2 = %d\n", lucro\_M2);\*/

}

}

if(aux == J\_1){

printf("\n");

aux=0;

}

printf("%d \t", matriz\_1[i][j]);

aux++;

}

printf("Lucro no mes %d = %d", i+1, (lucro\_M1+lucro\_M2));

}

printf("\n\nLucro anual total: %d", (tot\_1+tot\_2));

printf("\n\n");

printf("TABELA 2:");

for(i=0; i<I\_2; i++){

for (j=0; j<J\_2; j++){

if(aux == J\_2){

printf("\n");

aux=0;

}

printf("%d \t", matriz\_2[i][j]);

aux++;

}

}

}

|  |
| --- |
| Exercício 8 |

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

#define N 30

void ordem(char []);

int main(){

char string[N];

printf("Insira a string:\n");

scanf("%s", string);

ordem(string);

}

void ordem(char palavra[N]){

int i,j, maiorValor=0;

for(i=0; i<strlen(palavra); i++){

for(j=i+1; j<strlen(palavra); j++){

if(palavra[i] > palavra[j]){

maiorValor = palavra[i];

palavra[i] = palavra[j];

palavra[j] = maiorValor;

}

}

}

printf("%s", palavra);

}

|  |
| --- |
| Exercício 9 |

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

#define N 30

char \*strchar (char \*s, char ch){

int i;

for(i=0; i<N; i++){

if(\*(s+i) == ch){

return (s+i);

}

}

return NULL;

}

int main(){

char s[N], ch='a';

printf("Insira a string:\n");

scanf("%s", s);

printf("\nValor de s: %s\n", s);

if(strchar(s, ch) == NULL){

printf("A string s nao contem o caractere %c!!!", ch);

}else{

printf("\nEndereco: %p", strchar(s, ch));

}

}

|  |
| --- |
| Exercício 10 |

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

#define N 30

char \*strchar (char \*s, char ch){

int i;

for(i=N; i>0; i--){

if(\*(s+i) == ch){

return (s+i);

}

}

return NULL;

}

int main(){

char s[N], ch='a';

printf("Insira a string:\n");

scanf("%s", s);

printf("\nValor de s: %s\n", s);

if(strchar(s, ch) == NULL){

printf("A string s nao contem o caractere %c!!!", ch);

}else{

printf("\nEndereco: %p", strchar(s, ch));

}

}

|  |
| --- |
| Exercício 11 |

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

#define N 70

void imprime\_sobrenome (char \*nome\_completo){

int a,i, cont=0;

char sobrenome[N];

a = strlen(nome\_completo);

for(i=a-1; \*(nome\_completo+i)!= ' '; i--){

sobrenome[cont]=nome\_completo[i];

cont++;

}

printf("\nSobrenome: %s", strrev(sobrenome));

}

int main(){

char s[N];

printf("Digite seu nome completo:\n");

scanf("%[^\n]", s);

printf("\nNome Completo: %s\n", s);

imprime\_sobrenome(s);

}

|  |
| --- |
| Exercício 12 |

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

#define N 30

#define M 10

char \*stringstring (char \*str1, char \*str2){

int i=0, j, m , n;

n = strlen(str1);

m = strlen(str2);

if((strlen(str1))>= (strlen(str2))){

for(i=0; i<strlen(str1); i++){

if(\*(str1+i) == \*(str2+i)){

return (str1+i);

}

}

}else{

return NULL;

}

}

int main(){

char str1[N], str2[M], \*endereco;

printf("Digite a string 1:\n");

scanf("%s", str1);

printf("Digite a string 2:\n");

scanf("%s", str2);

if ((stringstring(str1, str2)) == NULL){

printf("String 1 menor que a string 2!!!\n");

}else{

if((stringstring(str1,str2)) != NULL){

printf("Endereco onde str1 = str2: %x", stringstring(str1, str2));

}

else{

printf("String 2 nao pertence a string 1!!!\n");

}

}

}

|  |
| --- |
| Exercício 13 |

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

#define N 10

void calc(int \* v, int num, int \* xmin, int \* xmax){

int k;

for(k=0; k<num; k++){

if(\*(v+k) > \*xmax){

\*xmax = \*(v+k);

}

if(\*(v+k) < \*xmin){

\*xmin = \*(v+k);

}

}

}

int main(){

int v[N], num, xmin=0, xmax=0, i;

for(i=0; i<N; i++){

v[i] = rand()%200;

}

for(i=0; i<N; i++){

printf("%d\t", v[i]);

}

num = rand()%10;

printf("\nNUM = %d", num);

xmin = v[0];

xmax = v[0];

calc(v, num, &xmin, &xmax);

printf("\n\nO maior valor: %d", xmax);

printf("\nO menor valor: %d", xmin);

}

|  |
| --- |
| Exercício 14 |

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

#define N 70

void eliminar(char \*v, char ch){

int i,j, aux=0, cont=0;

for(i=0; i<strlen(v); i++){

if(\*(v+i) == ch){

\*(v+i) = ' ';

cont++;

}

}

for(i=0; i<strlen(v); i++){

for(j=i+1; j<strlen(v); j++){

if(\*(v+i) == ' '){

\*(v+i) = \*(v+j);

}

}

}

printf("%s", v);

}

int main(){

char s[N], ch='a';

printf("Insira a string:\n");

scanf("%s", s);

printf("\nValor de s: %s\n", s);

eliminar(s, ch);

}