Nome: Gabriel Victorino

Exercicios

1) Interface gráfica do usuário, Texto, Aplicativo, Email

Descrição gerada automaticamente

2) Texto

Descrição gerada automaticamente com confiança média

3)

Interface gráfica do usuário, Texto, Aplicativo

Descrição gerada automaticamente

4)

Texto

Descrição gerada automaticamente

5)

Interface gráfica do usuário, Texto

Descrição gerada automaticamente

6)Tela de computador com texto preto sobre fundo branco

Descrição gerada automaticamente

Estrutura de Decisão

7)

Texto

Descrição gerada automaticamente

8)

Texto

Descrição gerada automaticamente

9)

Texto

Descrição gerada automaticamente

10)Texto

Descrição gerada automaticamente

11)

Texto

Descrição gerada automaticamente

12)

Texto

Descrição gerada automaticamente

13)

#include <iostream>

#include <string.h>

using namespace std;

//validador de cpf

/\* run this program using the console pauser or add your own getch, system("pause") or input loop \*/

int main(int argc, char\*\* argv) {

int soma=0, resultado = 0, digito10, digito11, digito=0;

char cpf[11];

cout << "Informe o cpf (Sem espaços e pontos). \n\n";

cin >> cpf;

if(strlen(cpf) != 11){

cout << "Cpf nao possui 11 numeros.\n";

return 0;

}else if(

(strcmp(cpf,"00000000000") == 0) || (strcmp(cpf,"11111111111") == 0) || (strcmp(cpf,"22222222222") == 0) ||

(strcmp(cpf,"33333333333") == 0) || (strcmp(cpf,"44444444444") == 0) || (strcmp(cpf,"55555555555") == 0) ||

(strcmp(cpf,"66666666666") == 0) || (strcmp(cpf,"77777777777") == 0) || (strcmp(cpf,"88888888888") == 0) ||

(strcmp(cpf,"99999999999") == 0)

){

cout << "Todos os numeros repetidos.\n\n";

return 0;

}else{

//digito verificador 1

for(int i = 0; i < 9; i++){

digito = cpf[i]-'0';

// cout << digito << endl;

soma+= digito \* (10 - i);

// cout << soma << endl;

}

digito10 = (soma \* 10) % 11;

if(digito10 == 10 || digito10 == 11){

digito10 = 0;

}

//digito verificador 2

soma = 0;

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

digito = cpf[i]-'0';

// cout << digito << endl;

soma+= digito \* (11 - i);

// cout << soma << endl;

}

digito11 = (soma \* 10) % 11;

if(digito11 == 10 || digito11 == 11){

digito10 = 0;

}

if(digito10 == cpf[9]-'0' && digito11 == cpf[10]-'0'){

cout << "CPF VALIDO!" << endl;

}else{

cout << "CPF INVALIDO" << endl;

}

}

return 0;

}

14)

#include <iostream>

#include <vector>

#include <algorithm>

#include <iterator> // Inclua este cabeçalho para usar ostream\_iterator

using namespace std;

int main() {

int NUM\_MAX = 0;

cout << "Informe um numero maximo: \n\n";

cin >> NUM\_MAX;

vector<int> vetor;

//popula o vetor

for(int i = 1; i <= NUM\_MAX; i++){

vetor.push\_back(i);

}

cout << endl;

// // Removendo o primeiro ultimo elemento

// vetor.erase(vetor.begin());

// vetor.pop\_back();

copy(vetor.begin(), vetor.end(), ostream\_iterator<int>(cout, " "));

cout << std::endl;

while(vetor.size() > 1){

// Removendo o primeiro ultimo elemento

vetor.erase(vetor.begin());

vetor.pop\_back();

copy(vetor.begin(), vetor.end(), ostream\_iterator<int>(cout, " "));

cout << std::endl;

}

return 0;

}

15)

#include <iostream>

#include <vector>

#include <algorithm>

#include <iterator> // Inclua este cabeçalho para usar ostream\_iterator

using namespace std;

/\* run this program using the console pauser or add your own getch, system("pause") or input loop \*/

int main(int argc, char\*\* argv) {

int max = 21;

int termo\_a, termo\_b, termo\_c;

int i\_a = 0,i\_b = 1;

vector<int> vetor;

cout << "Fibonacci" << endl;

vetor.push\_back(1);//e 1

vetor.push\_back(1);//e 2

termo\_a = vetor[0];

termo\_b = vetor[1];

while(vetor.size() < 8){

termo\_a = vetor[i\_a++];

termo\_b = vetor[i\_b++];

termo\_c = termo\_a + termo\_b;

vetor.push\_back(termo\_c);

cout << "A: " << termo\_a << " - i\_a= " << i\_a << endl;

cout << "B: " << termo\_b << " - i\_b= " << i\_b << endl;

cout << "Tamanho vetor: " << vetor.size() << endl;

copy(vetor.begin(), vetor.end(), ostream\_iterator<int>(cout, ", "));

cout << std::endl;

cout << "\*\*\*\*\n";

}

return 0;

}

16)

#include <iostream>

#include <math.h>

using namespace std;

/\* run this program using the console pauser or add your own getch, system("pause") or input loop \*/

int main(int argc, char\*\* argv) {

unsigned long long int total;

unsigned long long int qtde\_atual;

int MAX\_QUADROS = 64;

for(int i = 1; i <= MAX\_QUADROS; i++){

qtde\_atual = pow(2, i - 1); //Observa-se que o dobro (2) a cada quadro (i)

total += qtde\_atual;

cout << "QUADRO: " << i << " GRAOS: " << qtde\_atual << endl;

}

cout << "TOTAL: "<< total << endl;

return 0;

}

17)

#include <iostream>

using namespace std;

/\* run this program using the console pauser or add your own getch, system("pause") or input loop \*/

int main(int argc, char\*\* argv) {

int opcao = 1;

double p1,p2,p3, pagamento, total, troco;

while(opcao != 0){

"SAIR - 0\n CONTINUAR - 1\n\n";

cout << "Loja Tabajara: " << endl;

cout << "Informe o valor dos 3 produtos que comprou: " << endl;

cin >> p1 >> p2 >> p3;

total = p1 + p2 + p3;

cout << "Total: R$" << total << endl;;

cout << "Informe seu pagamento: " << endl;

cin >> pagamento;

if(pagamento < total){

cout << "Insira novamente o valor: " << endl;

cin >> pagamento;

}else{

troco = pagamento - total;

cout << "Seu troco será R$ " << troco << endl;

cout << "Deseja continuar? SAIR - 0\n CONTINUAR - 1\n\n";

cin >> opcao;

}

}

return 0;

}

18)

#include <iostream>

using namespace std;

/\* run this program using the console pauser or add your own getch, system("pause") or input loop \*/

int main(int argc, char\*\* argv) {

int opcao = 8; //1234

int total\_c1=0, total\_c2=0, total\_c3=0, total\_c4=0, total\_vb=0, total\_vn=0, total\_geral=0;

double p\_vn =0.00, p\_vb=0.00;

while(opcao != 0){

cout << "|----------------------------MENU------------------------|\n";

cout << "|CANDIDATO: 1--------------------------------------------|\n";

cout << "|CANDIDATO: 2--------------------------------------------|\n";

cout << "|CANDIDATO: 3--------------------------------------------|\n";

cout << "|CANDIDATO: 4--------------------------------------------|\n";

cout << "|VOTO NULO: 5--------------------------------------------|\n";

cout << "|VOTO BRANCO: 6------------------------------------------|\n";

cout << "|VER RESULTADOS(SAIR) - 0 ----------------------------|\n";

cout << "|--------------------------------------------------------|\n";

cout << "\n\n";

cout << ">> ";

cin >> opcao;

switch(opcao){

case 1:

total\_c1++;

break;

case 2:

total\_c2++;

break;

case 3:

total\_c3++;

break;

case 4:

total\_c4++;

break;

case 5:

total\_vb++;

break;

case 6:

total\_vn++;

break;

}

}

cout << "\n\n\n";

total\_geral = total\_c1 + total\_c2 + total\_c3 + total\_c4 + total\_vn + total\_vb;

p\_vb = (total\_vb / float(total\_geral) \* 100.00);

p\_vn = (total\_vn / float(total\_geral) \* 100.00);

cout << "BRANCOS " << p\_vb << "%\n";

cout << "NULOS " << p\_vn << "%\n";

system("pause");

return 0;

}

19)

#include <iostream>

#include <string.h> // cstring

using namespace std;

/\* run this program using the console pauser or add your own getch, system("pause") or input loop \*/

int main(int argc, char\*\* argv) {

int remover;

string txt="Algoritimo e Logica de Programacao";

cout << "Lembre-se que caracter a != A\n\n";

cout << "Informe a posicao do caracter a remover: \n";

cin >> remover;

if(remover < txt.length() && remover >= 0) {

txt = txt.erase(remover, 1);

}else{

cout << "Informe uma posicao valida.";

}

cout << "Nova string: " << txt << endl;

system("pause");

return 0;

}

20)

#include <iostream>

#include <string.h> // cstring

using namespace std;

/\* run this program using the console pauser or add your own getch, system("pause") or input loop \*/

int main(int argc, char\*\* argv) {

int adicionar;

char caracter = ' ';

string txt="Algoritimo e Logica de Programacao";

cout << "Lembre-se que caracter a != A\n\n";

cout << "Informe a posicao do caracter a adicionar e qual o caracter a ser adicionado: \n";

cin >> adicionar >> caracter;

if(adicionar < txt.length() && adicionar >= 0) {

txt = txt.insert(adicionar, 1, caracter);

}else{

cout << "Informe uma posicao valida.";

}

cout << "Nova string: " << txt << endl;

system("pause");

return 0;

}

21)

#include <iostream>

#include <vector>

using namespace std;

/\* run this program using the console pauser or add your own getch, system("pause") or input loop \*/

int main() {

vector<int> vetor;

int valor\_buscar;

int i = 0;

bool encontrado = false;

vetor.push\_back(5);

vetor.push\_back(10);

vetor.push\_back(4);

vetor.push\_back(15);

cout << "O vetor possui os seguintes valores: \n";

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

cout << vetor[i] << " ";

}

cout << endl;

cout << "Informe o valor que deseja buscar: \n";

cin >> valor\_buscar;

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

if(vetor[i] == valor\_buscar){

cout << "O vetor possui o valor procurado: " << vetor[i] << endl;

encontrado = true;

}else{

cout << "Essa posicao do vetor nao corresponde ao valor especificado." << endl;

encontrado == false;

}

}

return 0;

}

22)

#include <iostream>

#include <vector>

using namespace std;

/\* run this program using the console pauser or add your own getch, system("pause") or input loop \*/

int main(int argc, char\*\* argv) {

vector <int> vetor;

int n1, n2, n3, n4, n5, n6, n7, n8, n9, n10;

int valor\_referencia;

int contador=0;

cout << "Informe 10 numeros inteiros\n";

cin >> n1 >> n2 >> n3 >> n4 >> n5 >> n6 >> n7 >> n8 >> n9 >> n10;

vetor.push\_back(n1);

vetor.push\_back(n2);

vetor.push\_back(n3);

vetor.push\_back(n4);

vetor.push\_back(n5);

vetor.push\_back(n6);

vetor.push\_back(n7);

vetor.push\_back(n8);

vetor.push\_back(n9);

vetor.push\_back(n10);

cout << "\n";

//exibe vetor completo

cout << "Vetor completo.\n";

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

cout << vetor[i] << " ";

}

cout << endl;

cout << "Informe um valor de referencia.\n";

cin >> valor\_referencia;

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

if(vetor[i] > valor\_referencia){

cout << valor\_referencia << " e menor que " << vetor[i] << endl;

}

}

//menor que

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

if(vetor[i] < valor\_referencia){

cout << valor\_referencia << " e maior que " << vetor[i] << endl;

}

}

//quantas vezes aparece no vetor

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

if(vetor[i] == valor\_referencia){

contador++;

}

}

cout << "O valor " << valor\_referencia << " aparece " << contador << "vezes no vetor.\n";

return 0;

}

23)

#include <iostream>

#include <vector>

using namespace std;

/\* run this program using the console pauser or add your own getch, system("pause") or input loop \*/

int main(int argc, char\*\* argv) {

vector <int> vetor;

for(int i = 0;i < 100;i++){

if(i % 7 != 0 && i % 10 != 7){

vetor.push\_back(i);

}

}

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

cout << vetor[i] << " ";

}

return 0;

}

24)

#include <iostream>

#include <vector>

using namespace std;

/\* run this program using the console pauser or add your own getch, system("pause") or input loop \*/

int main(int argc, char\*\* argv) {

vector <int> vetor;

int inserir\_numero = 0;

bool numero\_validado = true;

int indice\_vetor, indices\_anteriores;

/\*

vetor.push\_back(1);

vetor.push\_back(2);

vetor.push\_back(3);

vetor.push\_back(4);

vetor.push\_back(5);

vetor.push\_back(6);

vetor.push\_back(7);

vetor.push\_back(8);

vetor.push\_back(9);

vetor.push\_back(10);

\*/

for(indice\_vetor =0;indice\_vetor<10;indice\_vetor++){

cout << "Informa numero: \n";

cin >> inserir\_numero;

for(indices\_anteriores = 0; indices\_anteriores < indice\_vetor;indices\_anteriores++){

if(inserir\_numero == vetor[indices\_anteriores]){

numero\_validado = false;

cout << "Numero reptido;\n";

indice\_vetor--;

break;

}

}

if(numero\_validado == true){

vetor.push\_back(inserir\_numero);

}else{

numero\_validado = true; // ele ativa o true para poder repetir o for e revalidar o novo numero

}

}

cout << "Vetor completo: " << endl;

for(indice\_vetor = 0; indice\_vetor < 10;indice\_vetor++){

cout << vetor[indice\_vetor] << " ";

}

return 0;

}

25)

#include <iostream>

#include <locale.h>

using namespace std;

int main()

{

    setlocale(LC\_ALL,"");

    //matriz transposta Ã© a troca de linhas por colunas

    //linha / coluna

    int MAX\_LINHAS = 2;

    int MAX\_COLUNAS = 3;

    int l, c;

    int matriz[MAX\_LINHAS][MAX\_COLUNAS];

    matriz[0][0] = 1;

    matriz[0][1] = 2;

    matriz[0][2] = 3;

    matriz[1][0] = 4;

    matriz[1][1] = 5;

    matriz[1][2] = 6;

    cout << "Matriz normal: \n";

    for(l = 0; l < MAX\_LINHAS;l++){//renderiza linhas

        for(c = 0; c < MAX\_COLUNAS; c++){//renderiza colunas

            cout << matriz[l][c] << " ";

        }

        cout << endl;

    }

    cout << endl;

    cout << "Matriz transposta: \n";

    for(c = 0; c < MAX\_COLUNAS; c++){//renderiza colunas

        for(l = 0; l < MAX\_LINHAS;l++){//renderiza linhas

            cout << matriz[l][c] << " ";

        }

            cout << endl;

    }

    return 0;

}

26)

#include <iostream>

#include <stdlib.h>

#include <locale.h>

#include <ctime>

using namespace std;

int main()

{

    setlocale(LC\_ALL,"");

    //matriz transposta é a troca de linhas por colunas

    //linha / coluna

    int MAX\_LINHAS = 10;

    int MAX\_COLUNAS = 10;

    int local\_l = 0, local\_c = 0;

    int l, c;

    int valor;

    int matriz[MAX\_LINHAS][MAX\_COLUNAS];

    int MAIOR;

    srand(time(0));

    for(l = 0; l < MAX\_LINHAS;l++){//renderiza linhas

        for(c = 0; c < MAX\_COLUNAS; c++){//renderiza colunas

           valor = rand() % 10000 + 1;

           matriz[l][c] = valor;

        }

        cout << endl;

    }

    cout << "Matriz 10x10: \n";

    for(l = 0; l < MAX\_LINHAS;l++){//renderiza linhas

        for(c = 0; c < MAX\_COLUNAS; c++){//renderiza colunas

            cout << " " << matriz[l][c] << "   ";

        }

        cout << endl;

    }

    cout << endl;

    MAIOR = matriz[0][0];

    for(l = 0; l < MAX\_LINHAS;l++){

        for(c = 0; c < MAX\_COLUNAS; c++){

           valor = matriz[l][c];

           if(valor > MAIOR){

               MAIOR = valor;

               local\_l = l + 1;

               local\_c = c + 1;

           }

        }

    }

    cout << "O maior numero e: " << MAIOR << " localizado em " << local\_l << " x " << local\_c << endl;

    return 0;

}

27)

#include <iostream>

#include <stdlib.h>

#include <locale.h>

#include <ctime>

using namespace std;

int main()

{

setlocale(LC\_ALL,"");

//linha / coluna

int MAX\_LINHAS = 24;

int MAX\_COLUNAS = 4;

int l, c;

int valor, total;

int media = 0, MAIOR = 0, paciente=0;

int matriz[MAX\_LINHAS][MAX\_COLUNAS];

srand(time(0));

for(l = 0; l < MAX\_LINHAS;l++){//renderiza linhas

for(c = 0; c < MAX\_COLUNAS; c++){//renderiza colunas

valor = rand() % 100 + 1;

matriz[l][c] = valor;

}

cout << endl;

}

cout << "Matriz 24x4: \n";

for(l = 0; l < MAX\_LINHAS;l++){//renderiza linhas

for(c = 0; c < MAX\_COLUNAS; c++){//renderiza colunas

cout << " " << matriz[l][c] << " ";

}

cout << endl;

}

cout << endl;

cout << "Médias: \n";

for(c = 0; c < MAX\_COLUNAS; c++){//renderiza colunas

for(l = 0; l < MAX\_LINHAS;l++){//renderiza linhas

total += matriz[c][l];

}

cout << endl;

media = total / 24;

if(MAIOR < media){

MAIOR = media;

paciente = c + 1;

}

cout << "Média Paciente " << c + 1 << " " << total / 24 << endl;

total = 0;

}

cout << "Maior: " << MAIOR << " - Paciente: " << paciente << endl;

return 0;

}

28)

29)

#include <iostream>

#include <locale.h>

using namespace std;

int retorna\_posicao(char c, int size\_string, string texto){

    int posicao = 0;

    for(int i = 0; i < size\_string;i++){

        if(texto[i] == c){

            posicao = i;

            break;

        }

    }

    return posicao;

}

int main()

{

    setlocale(LC\_ALL,"");

    string texto;

    char caracter;

    cout << "Informe um texto (sem acentos): \n";

    cin >> texto;

    cout << "Informe um caracter: \n";

    cin >> caracter;

    cout << "A posicao de " << caracter << " e: " << retorna\_posicao(caracter, texto.length(), texto) + 1;

    return 0;

}