|  |  |  |
| --- | --- | --- |
| logo_CEFSA_H(grande) | **Código: <B 10102>**  **Disciplina: <Algoritmos I>** N1 | 2º bimestre| Curso: EC | Turma: 114/11/2022 Prof.: Flávio Viotti | Coord.: Rodrigo Tadeu Fontes | **Logo Faculdade Eng** |
| Aluno(a):­­­­­­­­­­­­­­­­­­­­­­­­­­­­­­­­­­­­­­­­­­­­ Nº: RA: | | Nota: |
| Orientações: Cole o código do seu programa abaixo de cada uma das questões | |
|  | | Rubrica do aluno: |

**Lista de Exercícios**

Para todos os exercícios, faça o programa em C#. Necessário somente o arquivo ‘program.cs’ da sua resolução, todos os exercícios deverão estar em sequencia, caso algum exercício não seja feito, informe no lugar dele a frase “NÃO RESOLVIDO”.

Escolha somente 10 exercícios dentre os 15 disponíveis. Para os programas que não foram escolhidos, por favor deixa-los com a cor do enunciado em **vermelho**.

Utilize para os programas a fonte Courier New tamanho 9 ou 10, ou use o programa Notepad++ para sua impressão. **Certifique-se de que não haverá linhas em branco entre uma linha de código e outra**. Trabalhos que não cumpram essas regras não serão considerados. Exemplo

Program teste;

CORRETO

Uses CRT;

Var

X,z,y :integer;

Program teste;

Uses CRT;

INCORRETO

Var

X,y,z : integer;

1. Para que a divisão entre 2 números possa ser realizada, o divisor não pode ser nulo (zero). Escreva um programa para ler 2 valores e imprimir o resultado da divisão do primeiro pelo segundo. OBS: O programa deve validar a leitura do segundo valor (que não deve ser nulo). Enquanto for fornecido um valor nulo a leitura deve ser repetida. Utilize a estrutura **While** na construção da repetição de validação.

//declaração da váriavel que controla o laço

char saiLaco = 'N';

//seleciona os dados para o usuário

Console.Write("Digite o primeiro número: ");

double num1 = Convert.ToDouble(Console.ReadLine());

Console.Write("Digite o segundo número: ");

while (saiLaco == 'N') {

double num2 = Double.Parse(Console.ReadLine());

if(num2 != 0)

{

saiLaco = 'S';

//efetua o calculo

double resul = num1 / num2;

Console.Write($" o resultado da divisão de {num1} por {num2} é {resul} ");

}

else

{

}

}

Console.ReadKey();

1. Altere a solução do exercício anterior para que seja impressa a mensagem **Valor inválido!** caso o segundo valor informado seja **zero**.

//declaração da váriavel que controla o laço

char saiLaco = 'N';

//seleciona os dados para o usuário

Console.Write("Digite um número: ");

double num1 = Convert.ToDouble(Console.ReadLine());

Console.Write("Digite outro número: ");

while (saiLaco == 'N') {

double num2 = Double.Parse(Console.ReadLine());

if(num2 == 0)

{

Console.Write("Valor inválido!: ");

}else

{

saiLaco = 'S';

//efetua o calculo

double resul = num1 / num2;

Console.Write($" o resultado da divisão de {num1} por {num2} é {resul} ");

}

}

Console.ReadKey();

1. Reescreva o programa para o exercício 1 utilizando a estrutura **do while** na construção da repetição de validação.

//declaração da váriavel que controla o laço

char saiLaco = 'N';

//seleciona os dados para o usuário

Console.Write("Digite o primeiro número: ");

double num1 = Convert.ToDouble(Console.ReadLine());

do {

Console.Write("Digite o segundo número: ");

double num2 = Double.Parse(Console.ReadLine());

if(num2 != 0)

{

saiLaco = 'S';

//efetua o calculo

double resul = num1 / num2;

Console.Write($" o resultado da divisão de {num1} por {num2} é {resul} ");

}

}while (saiLaco == 'N') ;

Console.ReadKey();

1. Escreva um programa para ler as notas da 1ª e 2ª avaliações de um aluno, calcular e imprimir a média semestral. Faça com que o algoritmo só aceite notas válidas (uma nota válida deve pertencer ao intervalo [0,10]). Cada nota deve ser validada separadamente. Deve ser impressa a mensagem **"Nota inválida"** caso a nota informada não pertença ao intervalo [0,10].

//solicita os dados da n1

Console.Write("Digite a nota da 1° prova: ");

n1 = Convert.ToDouble(Console.ReadLine());

if (n1 < 0.0 || n1 > 10.0)

{

Console.Write("Valor inválido, tente novamente. ");

}

else

{

saiLaco1 = 'S';

}

} while (saiLaco1 == 'N');

do

{

//solicita os dados da n1

Console.Write("Digite a nota da 2° prova: ");

n2 = Convert.ToDouble(Console.ReadLine());

if (n2 < 0.0 || n2 > 10.0)

{

Console.Write("Valor inválido, por favor digite novamente: ");

}

else

{

saiLaco2 = 'S';

}

} while (saiLaco2 == 'N');

//calcula a media

double result = (n1 + n2) / 2;

//imprime a media na tela

Console.WriteLine($"Sua média semestral é {result}");

Console.ReadLine();

1. Reescreva o programa para o exercício 4 para que no final seja impressa a mensagem **Novo cálculo (1.sim 2.não)** solicitando ao usuário que informe um código (1 ou 2) indicando se ele deseja ou não executar o programa novamente. Se for informado o código 1 deve ser repetida a execução de todo o programa para permitir um novo cálculo, caso contrário ele deve ser encerrado.

//declara dado do laço

char saiLaco1 = 'N', saiLaco2 = 'N';

double n1, n2, escolha;

Int16 terminaLaco = 1, invalido=1;

while(terminaLaco == 1){

do

{

//solicita os dados da n1

Console.Write("Digite a nota da 1° prova: ");

n1 = Convert.ToDouble(Console.ReadLine());

if (n1 < 0.0 || n1 > 10.0)

{

Console.Write("Valor inválido, tente novamente. ");

}

else

{

saiLaco1 = 'S';

}

} while (saiLaco1 == 'N');

do

{

//solicita os dados da n1

Console.Write("Digite a nota da 2° prova: ");

n2 = Convert.ToDouble(Console.ReadLine());

if (n2 < 0.0 || n2 > 10.0)

{

Console.Write("Valor inválido, por favor digite novamente: ");

}

else

{

saiLaco2 = 'S';

}

} while (saiLaco2 == 'N');

//calcula a media

double result = (n1 + n2) / 2;

//imprime a media na tela

Console.WriteLine($"Sua média semestral é {result}");

//imprime a mensagem para sair do laço

Console.Write("Novo Calculo? Digite 1 para sim e 2 para não: ");

do {

escolha = Convert.ToDouble(Console.ReadLine());

if(escolha != 2)

{

if(escolha != 1)

{

Console.Write("Valor digitado é inválido!");

}else

{

invalido=2;

}

}else

{

terminaLaco = 2;

}

}while(invalido == 1);

}

Console.ReadLine();

1. Escreva um programa que verifique a validade de uma senha fornecida pelo usuário. A senha válida é a palavra **teste** (em minúsculo). *Observação:* Se a senha informada pelo usuário for inválida, a mensagem **"ACESSO NEGADO"** deve ser impressa e repetida a solicitação de uma nova senha até que ela seja válida. Caso contrário deve ser impressa a mensagem **"ACESSO PERMITIDO"** junto com um número que representa quantas vezes a senha foi informada.

//declaração de dados

string senhaCorreta="teste", senhaInserida, liberaAcesso="negado";

//Solicita a senha para o usuário

Console.Write("Digite a senha: ");

//faz a condição de confirmação da senha

while (liberaAcesso == "negado") {

senhaInserida = Console.ReadLine();

if (senhaInserida != senhaCorreta)

{

Console.Write("ACESSO NEGADO \nTente novamente: ");

} else

{

Console.Write("ACESSO PERMITIDO ");

liberaAcesso = "liberado";

}

}

Console.ReadKey();

1. Escreva um programa para imprimir as letras de A a Z. Procure a função que imprima o código ASC de um numero.

//imprime na tela o alfabeto de A a Z

Console.WriteLine("Abaixo o alfabeto de 'A' á 'Z' e seus correspondentes em ASCII;");

for(char i = 'a'; i <= 'z'; i++)

{

Console.WriteLine("{0} : {1}",i,(int)i );

}

Console.ReadKey();

1. Escreva um programa que calcule o fatorial de N (N!), sendo que o valor inteiro de N deve ser escolhido pelo usuário. Sendo que:

N! = 1 \* 2 \* 3 \* ... \* (N - 1) \* N

0! = 1 *(por definição)*

//solicita o dado de entrada do usuario

Console.Write("Digite um número: ");

Int32 num = Convert.ToInt32(Console.ReadLine());

//lógica do qualculo da fatorial

Int32 calculoFatorial=1;

for(Int32 i = 1;num >1; num--)

{

calculoFatorial = calculoFatorial \* num;

}

Console.Write("O calculo Fatorial do número digitado é: {0}",calculoFatorial);

Console.ReadKey();

1. Escreva um programa para determinar e escrever a soma dos números pares de 100 a 200, inclusive.

Int32 somaDosNumerosPares = 0, quantNumerosPares=0;

for(Int32 num=100; num <= 200; num++)

{

if (num % 2 == 0)

{

somaDosNumerosPares = somaDosNumerosPares + num;

quantNumerosPares++;

}

}

//escreve na tela a soma dos números pares

Console.Write("Soma dos números pares de 100 a 200 é {0} \nsão um total de {1} numeros pares", somaDosNumerosPares, quantNumerosPares);

Console.ReadKey();

1. Faça um programa para ler a altura e o sexo (feminino, masculino) de 10 pessoas. Calcular e escrever:

- a maior e a menor altura

- a média de altura das mulheres

- o número de homens

//declaração de variavel

double maiorAltura = 0, menorAltura = 10.0, alturaFem = 0, mediaAlturaFem ;

Int32 sexoFem = 0, quantHom ;

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

{

Console.Write("Por favor, digite sua altura: ");

double alturaUsuario = Convert.ToDouble(Console.ReadLine());

Console.Write("Digite seu sexo (M-Masculino / F-Feminina): ");

char sexo = Console.ReadLine().ToUpper()[0];

if(alturaUsuario > maiorAltura)

{

maiorAltura = alturaUsuario;

}

if(alturaUsuario < menorAltura)

{

menorAltura = alturaUsuario;

}

if(sexo == 'F')

{

sexoFem++;

alturaFem = alturaFem + alturaUsuario;

}

}

//calculo da média de altura das mulheres

mediaAlturaFem = alturaFem / sexoFem;

//soma dos homens

quantHom = 10 - sexoFem;

//escreve na tela o resultado

Console.Write("Das 10 pessoas cadastradas a maior altura é {0:0.00}, a menor é {1:0.00}\na média da altura das mulher é {2:0.00} e a quantidade de homens é {3}", maiorAltura, menorAltura, mediaAlturaFem, quantHom);

Console.ReadKey();

1. Faça um programa que leia 10 números inteiros positivos, calcule e imprima os que são números perfeitos. Sendo que, um número perfeito é aquele cuja soma de seus divisores, exceto ele próprio, é igual ao número.

*Exemplo:* 6 é perfeito porque 1 + 2 + 3 = 6

//declara a variavel de contagem de numeros perfeitos

Int32 contagemNum=0;

//criação da lista onde vai ser armazenados os numeros perfeitos

Int32[] numerosPerfeitos = new Int32[10];

//laço que solicita os 10 numeros a serem digitados

for(Int32 i=0; i < 10; i++)

{

//declara a variavel de soma dos numeros divisores

Int32 somaDosNumeros = 0;

//Solicita 10 numeros inteiros positivos

Console.Write("Digite um número: ");

Int32 inteiroPositivos = Convert.ToInt32(Console.ReadLine());

//Criação de um laço que roda para identificar os numeros divisiveis inteiros

for(Int32 num=inteiroPositivos-1; num > 0; num--)

{

//condição que vê por quais numeros é divivel com resto 0

if(inteiroPositivos%num == 0)

{

somaDosNumeros = somaDosNumeros + num;

}

}

//condição que vê quais numeros são inteiros perfeitos

if(somaDosNumeros == inteiroPositivos)

{

contagemNum++;

numerosPerfeitos[contagemNum] = inteiroPositivos;

}

}

//laço para impressão dos dados armazenados na lista

foreach(Int32 p in numerosPerfeitos)

{

if(p != 0) {

Console.WriteLine($"Número perfeito: {p}");

}

}

Console.ReadKey();

1. Número primo é aquele que só é divisível por ele mesmo e por 1. Faça um programa que determine e escreva os números primos compreendidos entre 100 e 1000.

//declaração de variavel de contagem de numeros divisores

for(Int32 num=100;num <= 1000; num++)

{

Int32 contagemN = 0;

for (Int32 i = 1; i <= num; i++)

{

if (num % i == 0)

{

contagemN++;

}

}

if(contagemN == 2)

{

Console.WriteLine(num);

}

}

Console.ReadKey();

1. Faça um programa para ler o valor de uma coluna e, se válida, mostre uma "linha" vertical na coluna lida desenhada com caracteres \*.

//cria uma lista de numeros para verificação e validação dos numeros digitados pelo ususario

Int32[] listaNum = { 10, 20, 30, 40, 50, 60, 70, 80, 90, 100 };

//cria uma lista para guardar os numeros que conferem com a lista anterior

Int32[] novaListaNum = new Int32[10];

//criação de variável para contagem de numeros corretos

Int32 contNum = 0;

//Solicita ao usuário que insira a coluna de numeros a serem avaliados

Console.WriteLine("Digite abaixo uma coluna com 10 números inteiros: ");

//Criação da lógica condicional que compara os numeros da coluna

for(Int32 i=1; i <=10; i++)

{

Console.Write($"{i}°: ");

Int32 numDigitados = Convert.ToInt32(Console.ReadLine());

for(Int32 num=1;num<=listaNum.Length; num++)

{

if(numDigitados == listaNum[num-1]) {

contNum++;

novaListaNum[num - 1] = numDigitados;

}

}

}

//criação do laço que valida se todos os numeros estiverem certos

if(contNum == 10)

{

Console.WriteLine("\nCorreto, coluna válidada!");

for(Int32 b=0; b< novaListaNum.Length; b++)

{

Console.WriteLine($"{novaListaNum[b]} \*");

}

}

Console.ReadKey();

1. Criar um programa que apresente a sequencia de Fibonacci ate o numero de termos escolhido pelo usuário. Sendo a sequencia de Fibonacci: 1 1 2 3 5 8 13 21 ....

//solicita ao usuário que insira o número maximo para fibonacci

Console.Write("Digite o número inteiro máximo para contagem de Fibonacci: ");

Int32 seqFibo = Convert.ToInt32(Console.ReadLine());

//criação da variável de controle do laço

Int32 fibo = 1, a=0;

//criação da lógica de contagem de Fibonacci

while(fibo <= seqFibo)

{

Console.Write(a + " " + fibo + " ");

a += fibo;

fibo += a;

}

Console.Write(a);

Console.ReadKey();

1. Foi realizada uma pesquisa sobre algumas características físicas da população de certa região, a qual foram coletadas as seguintes informações referentes a cada habitante.

Sexo (M-masculino / F-feminino)

Cor dos Olhos (V-verdes / A-azuis / C-castanhos)

Cor dos Cabelos (L-louro / C-castanho / P-preto)

Idade

- Crie um programa que apresente a maior e menor idade da pesquisa, e apresente o percentual de indivíduos do sexo Feminino que tenham entre 18 e 35 anos, tenham cabelos Louros e olhos Verdes. A pesquisa foi realizada com 35 pessoas.