Lista 1 - 32. Escreva um programa que leia um caracter e diga se ele é uma vogal, consoante, número ou um símbolo (qualquer outro caracter, que não uma letra ou número).

Console.Write("Digite o número da sua agência: ");

*char* caracter = char.Parse(Console.ReadLine());

*char*[] vogais = {'a', 'e', 'i', 'o', 'u'};

*char*[] consoantes = {'b', 'c', 'd', 'f', 'g', 'h', 'j', 'k', 'l', 'm', 'n', 'p', 'q', 'r', 's', 't', 'v', 'w', 'x', 'y', 'z'};

*char*[] numeros = {'0', '1', '2', '3', '4', '5', '6', '7', '8', '9'};

*string* tipo = "";

foreach(*char* vogal in vogais) {

if (caracter == vogal) tipo = "Vogal";

}

foreach(*char* consoante in consoantes) {

if (caracter == consoante) tipo = "Consoante";

}

foreach(*char* numero in numeros) {

if (caracter == numero) tipo = "Número";

}

if (tipo == "") {

tipo = "Símbolo";

}

System.Console.WriteLine(tipo);

Lista 1 - 33. Escreva um algoritmo que leia 2 valores (x e y), que devem representar as coordenadas de um ponto em um plano. A seguir, determine qual o quadrante ao qual pertence o ponto, ou se está sobre um dos eixos cartesianos ou na origem (x=y=0).

Console.Write("Digite o X: ");

*float* x = float.Parse(Console.ReadLine());

Console.Write("Digite o Y: ");

*float* y = float.Parse(Console.ReadLine());

if (x == 0 && y == 0) {

Console.WriteLine("Na origem");

}

*else* if (x == 0 || y == 0) {

Console.WriteLine("Sobre um dos eixos cartesianos");

}

*else* if (x < 0) {

if (y > 0) {

Console.WriteLine("No segundo quadrante");

}

else {

Console.WriteLine("No terceiro quadrante");

}

}

*else* if (x > 0) {

if (y > 0) {

Console.WriteLine("No primeiro quadrante");

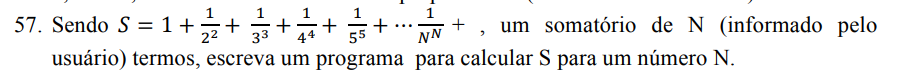
}

else {

Console.WriteLine("No quarto quadrante");

}

}

Lista 1 - 

Console.Write("Digite o N: ");

*int* n = int.Parse(Console.ReadLine());

*double* soma = 0;

for (*int* i = 1; i <= n; i++) {

*double* potencia = i;

for (*int* j = 1; j < i; j++) {

potencia \*= i;

}

soma += 1/potencia;

}

Console.WriteLine(soma);

Lista 2 - 52. Armazenar dez números na memória do computador. Exibir os valores na ordem inversa à da digitação.

*float*[] numeros = new *float*[10];

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

Console.Write("Digite o seu " + (i+1) + "o número: ");

numeros[i] = float.Parse(Console.ReadLine());

}

for (*int* i = 9; i >= 0; i--) {

Console.Write(numeros[i] + " ");

}

Lista 2 - 53. Armazenar dez valores na memória do computador. Após a digitação dos valores, criar uma rotina para ler os valores e achar e exibir o maior deles.

*float*[] numeros = new *float*[10];

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

Console.Write("Digite o seu " + (i+1) + "o número: ");

numeros[i] = float.Parse(Console.ReadLine());

}

*float* maior = numeros[0];

foreach (*float* numero in numeros) {

if (numero > maior) {

maior = numero;

}

}

Console.WriteLine("O maior valor é: " + maior);

Lista 2 - 54. Armazenar vinte valores em um vetor. Após a digitação, entrar com uma constante multiplicativa que deverá multiplicar cada um dos valores do vetor e armazenar o resultado no próprio vetor, na respectiva posição.

*float*[] numeros = new *float*[20];

const *float* multiplicativa = 3.2f;

for (*int* i = 0; i < 20; i++) {

Console.Write("Digite o seu " + (i+1) + "o número: ");

numeros[i] = float.Parse(Console.ReadLine());

}

for (*int* i = 0; i < 20; i++) {

numeros[i] = numeros[i] \* multiplicativa;

}

for (*int* i = 0; i < 20; i++) {

Console.WriteLine("O seu " + i + "o valor multiplicado é: " + numeros[i]);

}