**CÁC DẠNG BÀI TRÊN MẢNG:**

**Dạng 1: Nhập/xuất dãy số nguyên/thực.**

* Static void NhapMang (int []arrA);
* Static void XuatMang (int []arrA);
* Static void NhapMang (double []arrA);
* Static void XuatMang (double []arrA);

**Dạng 2: Đếm số lượng phần tử thỏa mãn điều kiện ABC**

* Static int DemABC (int/double []arrA, [//Co the khai bao them ABC])

{

int iDem = 0;

for(int i = 0; i < arrA.Length; i++)

{

if(arrA[i] la ABC…) // Vd KTSCP(arrA[i] == True)

{

iDem++;

}

}

return iDem;

}

* Lưu ý gọi main:

Console.WriteLine(“Kq = {0}”, DemABC (arrA));

* Nếu đếm số lần xuất hiện của X:
* Static int DemX (int/double []arrA, int/double X) {

int iDem = 0;

for(int i = 0; i < arrA.Length; i++)

{

if(arrA[i] == X)

{

iDem++;

}

}

return iDem;

}

* Lưu ý gọi main:

Int iX = 0;

//Nhap iX;

Console.WriteLine(“Kq = {0}”, demX (arrA, iX));

**Dạng 3: In ra các phần tử thỏa mãn điều kiện ABC.**

* Static void InABC (int/double []arrA, [//Co the khai bao them ABC])

{

for(int i = 0; i < arrA.Length; i++)

{

if(arrA[i] là ABC…)

{

Console.Write(“ {0} \t“, arrA[i]) ;

}

}

}

**Dạng 4: Tính tổng/tích/TBC các phần tử thỏa mãn điều kiện ABC.**

* int/double TinhABC(int/double []arrA) // trả về double nếu kết quả là số thực

{

int/ double nTong = 0; // int/ double nTich = 1;

for(int i = 0; i < arrA.Length; i++)

{

if(ABC…) // Nếu không có ĐK ABC thì bỏ if

{

nTong = nTong + arrA[i]; //nTich = nTich \* arrA[i];

}

}

return nTong; //return nTich; ….

}

* Lưu ý gọi main:

Console.WriteLine(“Kq = {0}”, tinhABC (arrA));

**Dạng 5: Tìm Max/ Min trong dãy số nguyên/thực**

Static int/double TimMax(int/double []arrA)

{

int iMax = arrA[0]; // double dMax = arrA[0];

for(int i = 0; i < arrA.Length; i++)

{

if(iMax < arrA[i]) //nếu tim Min thì thay dấu “<” bằng dấu “>”, đổi tên Max->Min

{

iMax = arrA[i];

}

}

return iMax;

}

* Lưu ý gọi main:

Console.WriteLine(“Max = {0}”, timMax(arrA));

**Dạng 5 – Nâng cao : Tìm Max/ Min là số thỏa mãn điều kiện ABC trong dãy.**

Int/float TimMaxABC(int/float []arrA)

{

Int/float nMax = So thoa ABC dau tien trong mang;

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

{

if(nMax < arrA[i] && arrA[i] là ABC)

//tim Min thay dấu “<” bằng “>”, đổi tên Max->Min

{

nMax = arrA[i];

}

}

return nMax;

}

* Lưu ý phần gọi hàm trong main()

Int kq = TimMaxABC(arrB,);

if (kq == INT\_MIN)

cout << “ Khong tim duoc so lon nhat thoa ABC”;

else

cout << “So Max thoa ABC la: “ << kq<< endl;

**Dạng 6: Kiểm tra mảng có tính chất ABC hay không?**

Static bool KiemTraMangABC(int []arrA)

{

for(int i = 0; i < arrA.Length; i++)

{

if ( ! ABC)

{

return false;

}

}

return true;

}

* Lưu ý phần gọi hàm trong main()

if (KiemTraMangABC(arrA) = = true)

Console.WriteLine( “ Mang co tinh ABC”);

else

Console.WriteLine( “Mang k co tinh ABC“);

**Dạng 7: Tìm phần tử thỏa ABC đầu tiên/ cuối cùng trong dãy. Tìm vị trí phần tử thỏa ABC đầu tiên/ cuối cùng trong dãy.**

Static int TimSoABC\_DauTien(int []arrA)

{

for(int i = 0; i < arrA.Length; i++) //Nếu cuối cùng: for(int i = arrA.Lenght-1; i >=0; i--)

{

if (arrA[i] thỏa ABC) //i thoa ABC

{

return arrA[i]; **//Nếu tìm vị trí thì return i;**

}

}

return -1;

}

* Lưu ý phần gọi hàm trong main()

int kq = TimSoABC\_DauTien (arrB);

if (kq = = -1)

Console.WriteLine( “ Khong tim duoc”);

else

Console.WriteLine( “So ABC dau tien la: {0}“, kq );

**Dạng 8: Sắp xếp mảng tăng/giảm dần.**

* Static void SapXepMangTang (int/double []arrA)

{

int iTam = 0; //neu so thuc : double dTam = 0.0d;

for(int i = 0; i < arrA.Length - 1; i++)

{

for(int j = i + 1; j < arrA.Length; j++)

{

if(arrA[i] > arrA[j]) // Nếu sx giảm dần: thay “>” bằng “<”.

{

iTam = arrA[i] ;

arrA[i] = arrA[j];

arrA[j] = iTam;

}

}

}

}

* Lưu ý: trong main(), sau khi gọi hàm sắp xếp phải gọi lại hàm XuatMang bên dưới mới xem được kết quả

**Dạng 9: Xóa phần tử tại vị trí k trong mảng.**

static void XoaViTri\_K(ref int[] arrA, int k)

{

if (k >= 0 && k <= arrA.Length - 1)

{

for (int j = k; j < arrA.Length - 1; j++)

{

arrA[j] = arrA[j + 1];

}

Array.Resize(ref arrA, arrA.Length - 1);

}

}

Lưu ý: trong main(), sau khi gọi hàm sắp xếp phải gọi lại hàm XuatMang bên dưới mới xem được kết quả

**Dạng 10: Thêm phần tử X sau vị trí k trong mảng.**

static void ThemXSau\_viTri(ref int[] arrA, int x, int vTri)

{

if (vTri >= 0 && vTri < arrA.Length)

{

Array.Resize(ref arrA, arrA.Length + 1);

for (int i = arrA.Length - 1; i > vTri; i--)

{

arrA[i] = arrA[i - 1];

}

arrA[vTri + 1] = x;

}

}

Lưu ý: trong main(), sau khi gọi hàm sắp xếp phải gọi lại hàm XuatMang bên dưới mới xem được kết quả

/\*

Ho ten: Nguyen Thi Thanh Thu

Lop: CD19TT7

MSSV: 19211TT4111

Mon thi: KTLT1

Ngay thi: 15/01/2022

Cau 2\*/

using System;

namespace Bai2

{

class Program

{

static void Main(string[] args)

{

//Khai bao bien

int n = 0;

char kyTu = ' ';

//Nhap du lieu

do

{

n = NhapSoNguyenDuong();

LKSNT(n);

Console.WriteLine("\n\nBan co muon tiep tuc chuong trinh");

kyTu = char.Parse(Console.ReadLine());

} while (kyTu == 'Y' || kyTu == 'y');

}

/// <summary>

/// Ham nhap so nguyen duong

/// </summary>

/// <returns></returns>

static int NhapSoNguyenDuong()

{

int n = 0;

do

{

Console.Write("Hay nhap so nguyen duong n: ");

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

} while (n <= 0);

return n;

}

/// <summary>

/// Kiem tra so nguyen to

/// </summary>

/// <param name="n"></param>

/// <returns></returns>

static int KTSNT(int n)

{

//Dem so luong uoc chung cua n

int count = 0;

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

{

if (n % i ==0)

{

count++;

}

}

//So sanh so luong UC voi 2

if (count == 2)

{

return 1;

}

else

{

return 0;

}

}

/// <summary>

/// Liet ke so nguyen to

/// </summary>

static void LKSNT(int n)

{

Console.WriteLine("Liet ke cac so nguyen to <= n");

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

{

if (KTSNT(i) == 1)

{

Console.Write(i + " ");

}

}

}

}

}

/\*

Ho ten: Nguyen Thi Thanh Thu

Lop: CD19TT7

MSSV: 19211TT4111

Mon thi: KTLT1

Ngay thi: 15/01/2022

Cau 3\*/

using System;

namespace Bai3

{

class Program

{

static void Main(string[] args)

{

//Khai bao bien

int n = 0, x = 0, nthem = 0, vtri = 0, k = 0;

int[] arr;

char kyTu = ' ';

//Nhap du lieu

do

{

//Nhap so luong phan tu mang

do

{

Console.Write("Nhap so luong phan tu mang: ");

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

} while (n < 1 || n > 100);

//Cap phat mang

arr = new int[n];

//Goi ham cau1: nhap mang

NhapMang(arr);

//Goi ham cau2: xuat mang

XuatMang(arr);

//Goi ham cau3: tinh tong so nguyen to trong mang

Console.WriteLine($"\nTong cac so nguyen to: {TinhTongSNT(arr)}");

//Goi ham cau4: tim vi tri cua gia tri X xuat hien dau tien trong mang

Console.Write("\nNhap so muon tim vi tri trong mang: ");

x = int.Parse(Console.ReadLine());

int vitri = XuatGTXXHDau(arr, x);

if (vitri != -1)

{

Console.WriteLine($"vi tri {x} xuat hien dau tien trong mang la {vitri}");

} else

{

Console.WriteLine($"Khong tim thay gia tri {x} trong mang");

}

//Goi ham cau7: Sap xep mang tang dan

Console.WriteLine("\nMang sau khi sap xep tang dan: ");

SapXepMangTang(arr);

XuatMang(arr);

//Goi ham cau8: Tinh tong cac gia tri chan trong mang

Console.Write($"\nTong cac so chan trong mang: {TinhTongSChan(arr)}");

//Goi ham cau5: Them phan tu vao mang tai mot vi tri nhap vao

Console.Write("\n\nNhap so muon them : ");

nthem = int.Parse(Console.ReadLine());

Console.Write("Nhap vi tri muon them: ");

vtri = int.Parse(Console.ReadLine());

ThemX(ref arr, nthem, vtri);

Console.WriteLine("\nMang sau khi them: ");

XuatMang(arr);

//Goi ham cau6: Xoa phan tu mang tai vi tri nhap vao

Console.Write("\nNhap vi tri muon xoa: ");

k = int.Parse(Console.ReadLine());

XoaViTri\_K(ref arr, k);

Console.WriteLine("\nMang sau khi xoa: ");

XuatMang(arr);

Console.WriteLine("\n\nBan co muon tiep tuc chuong trinh y/n");

kyTu = char.Parse(Console.ReadLine());

} while (kyTu == 'Y' || kyTu == 'y');

}

/// <summary>

/// Xuat mang

/// </summary>

/// <param name="arr"></param>

static void XuatMang(int[] arr)

{

for (int i = 0; i < arr.Length; i++)

{

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

}

Console.WriteLine();

}

/// <summary>

/// Nhap mang

/// </summary>

/// <param name="arr"></param>

static void NhapMang(int[] arr)

{

Random rd = new Random();

for (int i = 0; i < arr.Length; i++)

{

arr[i] = rd.Next(0, 50);

}

}

/// <summary>

/// Kiem tra so nguyen to

/// </summary>

/// <param name="n"></param>

/// <returns></returns>

static int KTSNT(int n)

{

//Dem so luong uoc chung cua n

int count = 0;

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

{

if (n % i == 0)

{

count++;

}

}

//So sanh so luong UC voi 2

if (count == 2)

{

return 1;

}

else

{

return 0;

}

}

/// <summary>

/// Tong cac so nguyen to

/// </summary>

/// <param name="arr"></param>

/// <returns></returns>

static int TinhTongSNT(int[] arr)

{

int sum = 0;

for (int i = 0; i < arr.Length; i++)

{

if (KTSNT(arr[i]) == 1)

{

sum = sum + arr[i];

}

}

return sum;

}

/// <summary>

/// Tim gia tri X xuat hien dau tien

/// </summary>

/// <param name="arr"></param>

/// <param name="x"></param>

/// <returns></returns>

static int XuatGTXXHDau(int[] arr, int x)

{

for (int i = 0; i < arr.Length; i++)

{

if (arr[i] == x)

{

return i;

}

}

return -1;

}

/// <summary>

/// Sap xep mang tang dan

/// </summary>

/// <param name="arr"></param>

static void SapXepMangTang(int[] arr)

{

int iTam = 0; //neu so thuc : double dTam = 0.0d;

for (int i = 0; i < arr.Length - 1; i++)

{

for (int j = i + 1; j < arr.Length; j++)

{

if (arr[i] > arr[j]) // Nếu sx giảm dần: thay “>” bằng “<”.

{

iTam = arr[i];

arr[i] = arr[j];

arr[j] = iTam;

}

}

}

}

/// <summary>

/// Kiem tra so chan

/// </summary>

/// <param name="n"></param>

/// <returns></returns>

static int KTSChan(int n)

{

if (n % 2 == 0)

{

return 1;

}

else

{

return 0;

}

}

/// <summary>

/// Tong cac so chan trong mang

/// </summary>

/// <param name="arr"></param>

/// <returns></returns>

static int TinhTongSChan(int[] arr)

{

int sum = 0;

for (int i = 0; i < arr.Length; i++)

{

if (KTSChan(arr[i]) == 1)

{

sum = sum + arr[i];

}

}

return sum;

}

/// <summary>

/// Them X giua mang

/// </summary>

/// <param name="arr"></param>

/// <param name="x"></param>

/// <param name="vtri"></param>

static void ThemX (ref int [] arr, int x, int vTri)

{

//Them o cuoi: arr.Length

if (vTri >= 0 && vTri < arr.Length -1)

{

Array.Resize(ref arr, arr.Length + 1);

for (int i = arr.Length - 1; i > vTri; i--)

{

arr[i] = arr[i - 1];

}

arr[vTri + 1] = x;

}

}

/// <summary>

/// Xoa phan tu tai vi tri k

/// </summary>

/// <param name="arrA"></param>

/// <param name="k"></param>

static void XoaViTri\_K(ref int[] arrA, int k)

{

if (k >= 0 && k <= arrA.Length - 1)

{

for (int j = k; j < arrA.Length - 1; j++)

{

arrA[j] = arrA[j + 1];

}

Array.Resize(ref arrA, arrA.Length - 1);

}

}

}

}

/\*

Ho ten: Nguyen Thi Thanh Thu

Lop: CD19TT7

MSSV: 19211TT4111

Mon thi: KTLT1

Ngay thi: 15/01/2022

Cau1\*/

using System;

namespace SoanThiKTLT

{

class Bai1

{

static void Main(string[] args)

{

//Khai bao bien

int a = 0, b = 0, c = 0, max, count =0;

//Nhap du lieu

do

{

Console.Write("Hay nhap so nguyen duong a: ");

a = int.Parse(Console.ReadLine());

} while (a <= 0);

do

{

Console.Write("Hay nhap so nguyen duong b: ");

b = int.Parse(Console.ReadLine());

} while (b <= 0);

do

{

Console.Write("Hay nhap so nguyen duong c: ");

c = int.Parse(Console.ReadLine());

} while (c <= 0);

//Kiem tra tinh chan le

if (a % 2 ==0 && b %2 ==0 && c%2 ==0)

{

Console.WriteLine("Ca ba deu la so chan");

}

else if (a % 2 != 0 && b % 2 != 0 && c % 2 != 0)

{

Console.WriteLine("Ca ba deu la so le");

}

else

{

if (a%2 == 0)

{

count++;

}

if (b%2 == 0)

{

count++;

}

if (c % 3 == 0)

{

count++;

}

Console.WriteLine("Co {0} so chan", count);

Console.WriteLine("Co {0} so le", 3 - count);

}

//Tim gia tri lon nhat

max = a;

if (max < b)

{

max = b;

}

if (max < c)

{

max = c;

}

Console.Write("Gia tri lon nhat la: {0}", max);

}

}

}

số cuối cung

static int KiemTraSCC (int[]arr, int x ) { int viTri = -1; for (int i = arr.Length - 1; i >= 0; i--) { if (arr[i] == x) { viTri = i; break; } } return viTri; }

static bool Shh(int n)

{

int b = 0;

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

{

if (n % i == 0)

{

b += i;

}

}

fibonaxi

static int Fi(int n) { int num1 = 1; int num2 = 1; int F = 1; for (int i = 2; i <=n; i++) { F = num1 + num2; num1 = num2; num2 = F; }

doi xung

static bool KTSDX(int a) { int tam = a, SoNguoc =0; while (tam > 0) { SoNguoc = SoNguoc \* 10 + tam % 10; tam = tam /10; } return (a == SoNguoc); } }

chinh phuong

static int Check(int n) { int s = (int)Math.Sqrt(n); if (s==Math.Sqrt(n)&& n>0) { return 1; } else return 0; } }

dem uoc so chung

static int SoUS(int n) {int dem =0 ; for (int i = 1; i<=n; i++) { if (n%i==0) { dem++; } } return dem; }