

Bài 3. Dữ liệu có cấu trúc, xử lý ngoại lệ (Thực hành)

Bài 1: Viết chương trình cho phép:

- + Người dùng nhập danh sách n số nguyên ($0 < n \leq 100$):
- + Tính tổng các phần tử chẵn, in ra.
- + Sắp xếp mảng tăng dần, in ra.

Chú ý: Viết trực tiếp trong **Main (không sử dụng phương thức) theo 2 cách: mảng và **List<T>***

Bài 2:(method) Viết chương trình giải phương trình bậc 2 $AX^2+BX+C=0$ (a, b, c là các hệ số nhập từ bàn phím). Với 2 phương thức: **Nhap** và **Giai**. Trong phương thức **Main** gọi 2 phương thức trên.

Bài 3:(method) Viết chương trình tính tiền điện gồm 2 Phương thức:

- Phương thức **Nhap** (nhập chỉ số đầu và cuối kiểu void có TS)
- Phương thức **TinhTien** (kiểu trả về giá trị có TS)
- Trong Phương thức **Main** gọi 2 Phương thức trên.

Biết rằng: 100 kw đầu giá là: 2000đ/kw

50 kw tiếp giá là: 2500đ/kw

50 kw tiếp giá là: 2800đ/kw

>200 kw giá là: 3500đ/kw

** Sử dụng **Try...Catch** để bắt lỗi nhập không đúng định dạng*

Bài 4 (dùng **List**): Viết chương trình gồm các **phương thức** sau:

- + Nhập vào một **List** các số thực. Dừng nhập khi nhập vào kí tự “k/K”.
- + Hiện thị danh sách vừa nhập lên màn hình.
- + Sắp xếp danh sách theo thứ tự tăng dần, hiện thị **List** ra.
- + Xóa các số âm trong danh sách, hiện thị **List** ra.
- + Nhập vào từ bàn phím một số x bất kỳ. Chèn x vào vị trí đúng thứ tự và hiện thị **List** đã bổ sung.
- + Ghi danh sách này ra file txt.
- + Đọc thông tin từ file và hiển thị lên màn hình console.

Trong **Main** gọi các phương thức trên.

Giải:

```
using System.Collections.Generic;
using System.IO;
class Program
{
    static void Main(string[] args)
    {
        List<double> list_t = new List<double>();
        Nhap(list_t);
        Console.WriteLine("List vừa nhập: ");
        HienThiList(list_t);
        TongChan(list_t);
        SapXepTang(list_t);
        XoaSoAm(list_t); //Tự Làm
        ChenX(list_t); //Tự Làm
        GhiFile(list_t);
        DocFile("E:\\realNum.txt");
        Console.ReadLine();
    }
}
```

```

static void Nhap(List<double> list_t)
{
    Console.WriteLine("Nhap cac so thuc (nhap k/K de dung:");
    string num = "";
    while (true)
    {
        num = Console.ReadLine();
        if (num == "k" || num == "K") break;
        list_t.Add(Convert.ToDouble(num));
    }
}

static void HienThiList(List<double> list_t)
{
    foreach (var element in list_t)
        Console.Write(element + " ");
    Console.WriteLine();
}

static void TongChan(List<double> list_t)
{
    double Tong = 0;
    for (int i = 0; i < list_t.Count; ++i)
        if (list_t[i] % 2 == 0)
        {
            Tong = Tong + list_t[i];
        }
    Console.WriteLine("Tong cac phan tu chan la:{0}", Tong);
}

static void SapXepTang(List<double> list_t)
{
    for (int i = 0; i < list_t.Count - 1; ++i)
        for (int j = i + 1; j < list_t.Count; j++)
            if (list_t[i] > list_t[j])
            {
                double temp = list_t[i];
                list_t[i] = list_t[j];
                list_t[j] = temp;
            }
    Console.WriteLine("List duoc sap xep tang dan: ");
    HienThiList(list_t);
}

static void GhiFile(List<double> list_t)
{
    StreamWriter W_realNum = new StreamWriter("E:\\realNum.txt");
    foreach (var element in list_t)
        W_realNum.Write(element + " ");
    W_realNum.Close();
    Console.WriteLine("\nDa ghi list vao file E:\\realNum.txt");
}

static void DocFile(string path)
{
    StreamReader R_realNum = new StreamReader(path);
    Console.WriteLine("List duoc doc tu file: ");
    while (R_realNum.Peek() != -1)
        {Console.Write(R_realNum.ReadLine()); }
    // Console.Write(R_realNum.ReadToEnd());
    R_realNum.Close();
    Console.WriteLine();
}
}

```