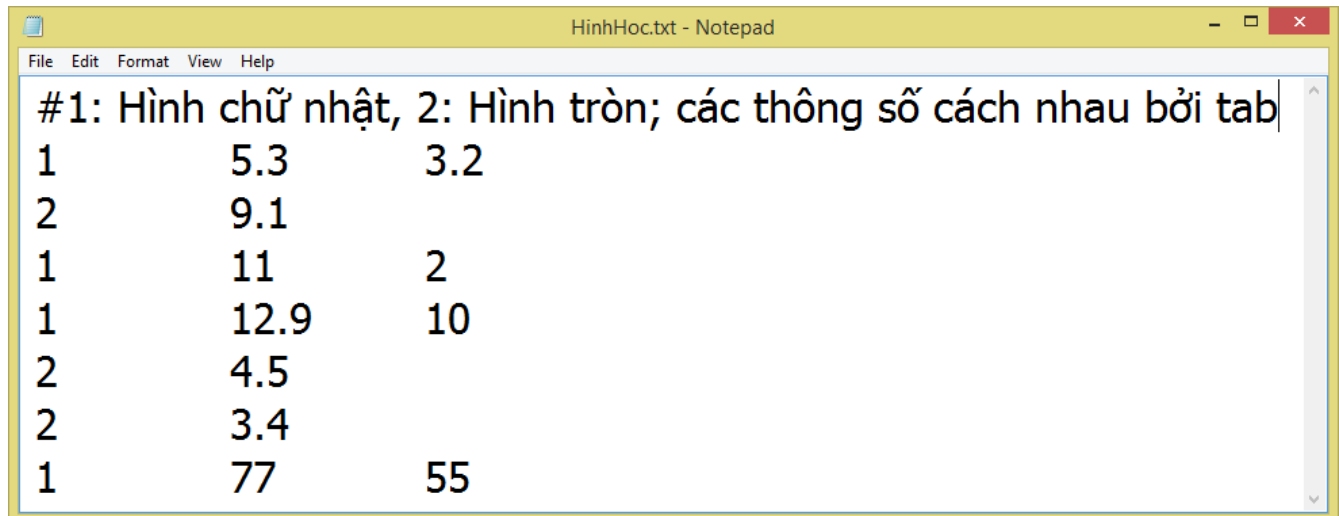


Hướng đối tượng với C#

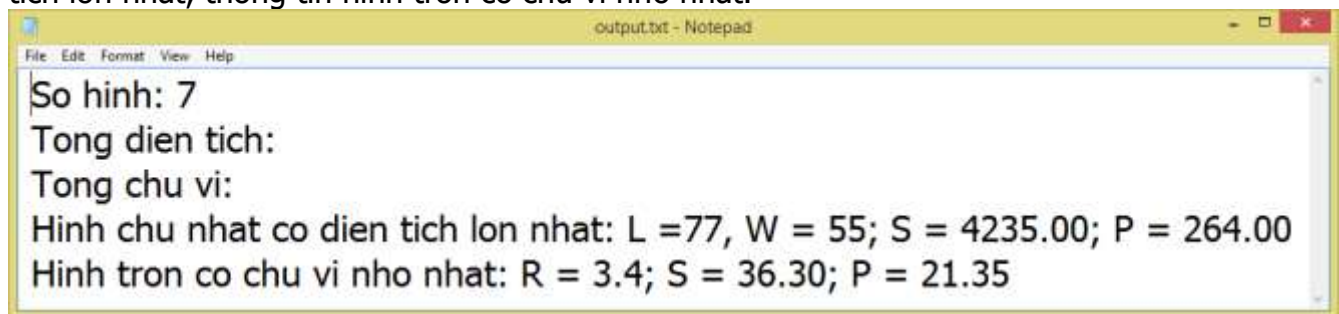
Bài tập 1: Quản lý Hình học

Tạo ứng dụng Console, thực hiện các yêu cầu sau:

- Thêm vào 1 class **HinhHoc** gồm các thành phần biểu diễn diện tích (DienTich) và chu vi (ChuVi).
 - Khai báo và định nghĩa các properties để truy cập đến giá trị của các thành phần dữ liệu diện tích, chu vi.
 - Viết phương thức **tinhDienTichChuVi()** để tính diện tích, chu vi.
 - Override phương thức **ToString()** để xuất diện tích, chu vi hình.
- Thêm vào 1 class **HinhChuNhat** kế thừa từ lớp HinhHoc biểu diễn thông tin hình chữ nhật bao gồm thuộc tính riêng của nó là mChieuDai, mChieuRong.
 - Khai báo thành phần dữ liệu cần thiết để biểu diễn hình chữ nhật
 - Khai báo và định nghĩa các constructor cần thiết để khởi tạo giá trị cho các thành phần dữ liệu.
 - Khai báo và định nghĩa các properties để truy cập đến giá trị của các thành phần dữ liệu mChieuDai và mChieuRong (get, set) .
 - Override phương thức **tinhDienTichChuVi()** để tính diện tích hình chữ nhật (mChieuDai x mChieuRong) và chu vi (mChieuDai+mChieuRong) x 2.
 - Override phương thức **ToString()** để xuất thông tin hình chữ nhật.
- Thêm 1 lớp tên **HinhTron** kế thừa từ lớp HinhHoc và viết thêm các thành phần sau :
 - Khai báo thêm thành phần dữ liệu bán kính: **mBanKinh**
 - Khai báo và định nghĩa các Constructor tham số và không tham số để khởi tạo các giá trị cho các thành phần dữ liệu.
 - Khai báo và định nghĩa các properties để truy cập đến giá trị của các thành phần dữ liệu (get, set).
 - Override phương thức **tinhDienTichChuVi()** để tính chu vi và diện tích hình tròn (cách viết giống như lớp hình chữ nhật).
 - Override phương thức **ToString()** để xuất thông tin hình tròn.
- Viết hàm Main() để kiểm tra các constructor, các properties, các phương thức của các lớp trên.
- Mở rộng, tạo List<HinhHoc> để quản lý thông tin các hình học. Đọc thông tin từ file text theo định dạng cho trước (tham khảo bài giảng **CS08 - File.pdf**) để lấy thông tin các hình. Cấu trúc file như sau:



Sau đó xuất ra file **output.txt** tổng diện tích, tổng chu vi, thông tin hình chữ nhật có diện tích lớn nhất, thông tin hình tròn có chu vi nhỏ nhất.



Gợi ý code:

```

//Lớp HinhHoc
public class HinhHoc
{
    #region AutomaticProperty
    public double DienTich { get; set; }
    public double ChuVi { get; set; }
    #endregion

    public virtual void tinhDienTichChuVi()
    {
    }
    public override string ToString()
    {
        return string.Format("S = {0}; P = {1}.", Math.Round(DienTich, 2), Math.Round(ChuVi, 2));
    }
}
  
```

//Lớp HìnhChuNhat

```
public class HìnhChuNhat: HìnhHoc
{
    //Fields
    private double mChieuDai;
    private double mChieuRong;

    //Property
    public double ChieuDai {
        get { return mChieuDai; }
        set
        {
            if (value > 0)
                mChieuDai = value;
            else
                throw new Exception("Chieu dai am");
        }
    }
    public double ChieuRong...

    //constructor
    public HìnhChuNhat(double d = 1, double r = 0.5)
    {
        ChieuDai = d; ChieuRong = r;
    }

    public override void tinhDienTichChuVi()
    {
        DienTich = ChieuDai * ChieuRong;
        ChuVi = (ChieuDai + ChieuRong) * 2;
    }

    public override string ToString()
    {
        return string.Format("L = {0}, W = {1}; {2}", ChieuDai, ChieuRong,
            base.ToString());
    }
}
} //end class
```

Bài tập 2: Quản lý Nhân viên

Tạo 1 ứng dụng Console, thực hiện các yêu cầu sau:

- Thêm vào 1 class tên *Ngươi* bao gồm các thành phần dữ liệu: *HoTen*, *NgaySinh*, *DiaChi*.
 - Khai báo và định nghĩa các constructor tham số và không tham số.

- Khai báo và định nghĩa các properties để truy cập đến giá trị của các thành phần dữ liệu (get, set) và viết thêm 1 properties chỉ đọc (get) dùng để lấy LayTuoi của Nguoi.
- Viết 1 phương thức tên XemThongTin(): xuất giá trị các thành phần dữ liệu ra màn hình.
- Thêm 1 lớp tên SinhVien kế thừa từ lớp Nguoi và viết thêm các thành phần sau:
 - Khai báo thêm các thành phần dữ liệu: string MaSV , string MaLop, string Email, string DienThoai.
 - Khai báo và định nghĩa các Constructor tham số và không tham số để khởi tạo các giá trị cho các thành phần dữ liệu .
 - Khai báo và định nghĩa các properties để truy cập đến giá trị của các thành phần dữ liệu (get, set) kiểm tra dữ liệu.
- Thêm 1 lớp tên NhanVien kế thừa từ lớp Nguoi và viết thêm các thành phần sau:
 - Khai báo thêm các thành phần dữ liệu: string MaNhanVien , string Email, string DienThoai, DateTime NgayLamViec, string MaCongTy.
 - Khai báo và định nghĩa các Constructor tham số và không tham số để khởi tạo các giá trị cho các thành phần dữ liệu.
 - Khai báo và định nghĩa các properties để truy cập đến giá trị của các thành phần dữ liệu (get, set) và kiểm tra dữ liệu.
- Viết hàm Main() để kiểm tra các constructor, các properties, các phương thức của các lớp trên.

Bài tập 3: Đa thức

Ta có định nghĩa đa thức bậc n như sau:

$$P = \sum_{k=0}^n a_k x^k$$

Xây dựng lớp **DaThuc** có các chức năng sau:

- Có hàm khởi tạo đa thức 1 tham số truyền vào là bậc của đa thức để tạo ra đa thức mà các hệ số đều bằng 0
- Có 2 phép toán là: phép cộng đa thức và phép trừ đa thức
- Có property cho biết bậc n của đa thức
- Có indexer với tham số truyền vào là số nguyên k cho biết giá trị a_k và có thể gán giá trị vào a_k
- Phương thức với 1 tham số truyền vào là b dùng để tính giá trị của đa thức khi $x = b$

---Hết---