

## BÀI THỰC HÀNH 2

### Nội dung:

- Định nghĩa lớp
- Sử dụng các đối tượng của lớp

**Bài 2.1** Định nghĩa lớp *Time* theo sơ đồ lớp và mô tả sau:

Time	
- hour: int - minute: int - second: int	- hour: phần giờ, minute: phút, second: giây - Time() phương thức khởi tạo không tham số, gán hour, minute, second bằng 0. - Time(int h, int m, int s) phương thức khởi tạo có tham số, gán hour bằng h, minute bằng m, second bằng s. - Minute, Second đóng gói các thuộc tính minute, second. - GetHour() trả về giá trị thuộc tính hour. - SetHour(int i) gán giá trị thuộc tính hour bằng i. - Show() hiện ra màn hình thời gian theo định dạng h:m:s
+ Time() + Time(int h, int m, int s) + Minute, Second + GetHour(): int + SetHour(int): void + Show(): void	

*Sử dụng lớp Time thực hiện các yêu cầu:*

- Tạo 2 đối tượng *Time* dùng 2 phương thức thiết lập khác nhau.
- Xuất thông tin 2 đối tượng ra màn hình.
- Thiết lập giá trị mới cho thuộc tính hour của đối tượng được tạo ra bằng phương thức thiết lập không tham số;
- Tính và in ra khoảng cách theo đơn vị giây (second) giữa hai mốc thời gian được lưu trong 2 đối tượng *Time*.

**Bài 2.2.** Định nghĩa lớp *Hình chữ nhật* gồm có:

- Các thuộc tính (Attributes) : chiều dài, chiều rộng.
- Các phương thức (Methods):
  - o Thiết lập không tham số gán giá trị chiều dài, chiều rộng là 0;
  - o Thiết lập có 2 tham số gán giá trị chiều dài, chiều rộng bằng giá trị 2 tham số truyền vào;
  - o Thiết lập sao chép gán giá trị chiều dài, chiều rộng bằng giá trị chiều dài, chiều rộng của đối tượng *Hình chữ nhật* truyền vào;
  - o Cập nhật giá trị của chiều dài, chiều rộng bằng các giá trị nhập vào từ bàn phím;
  - o Tính chu vi hình chữ nhật;
  - o Tính diện tích hình chữ nhật;

- Xuất ra màn hình thông tin hình chữ nhật gồm: chiều dài, chiều rộng, chu vi, diện tích.

*Sử dụng lớp Hình chữ nhật thực hiện các yêu cầu:*

- Tạo ra 3 đối tượng *Hình chữ nhật* dùng 3 phương thức khởi tạo khác nhau;
- In ra thông tin của 3 đối tượng vừa tạo;
- Tính và in ra tổng diện tích 3 hình chữ nhật.

**Bài 2.3.** Định nghĩa lớp Chuỗi gồm:

- Các thuộc tính: s (chuỗi s), x, y (vị trí xuất hiện của chuỗi s)
- Các phương thức:
  - Khởi tạo không tham số, khởi tạo có tham số: khởi tạo giá trị cho chuỗi s, vị trí xuất hiện chuỗi tại x, y, màu nền của cửa sổ, màu chữ.
  - Hiển thị: xóa màn hình, hiển thị chuỗi s ở vị trí x, y; ẩn con trỏ chuột.
  - Chuyển động: hiển thị chuỗi s ở vị trí x, y với màu chữ, màu nền mặc định trong phương thức khởi tạo; nếu bấm phím ← chuỗi s di chuyển sang trái, bấm phím → di chuyển sang phải, bấm phím ↓ di chuyển xuống dưới, bấm phím ↑ di chuyển lên trên, bấm phím 'Esc' dừng chương trình.

*Viết chương trình chính thực hiện:*

- Tạo 1 đối tượng chuỗi dùng phương thức thiết lập
- Thực hiện phương thức chuyển động của đối tượng.

*Ghi chú:* Sử dụng một số thuộc tính, phương thức của Console sau:

- Console.BackgroundColor = ConsoleColor.DarkBlue; // màu nền
- Console.ForegroundColor = ConsoleColor.White; // màu chữ gán là màu trắng
- Console.ForegroundColor = (ConsoleColor)c; //màu chữ là màu c=12
- Console.Clear();// xóa màn hình
- Console.CursorTop = y; // đưa con trỏ tới dòng y
- Console.CursorLeft = x; // đưa con trỏ tới cột x
- Console.CursorVisible = false; // ẩn con trỏ
- ConsoleKeyInfo phim = Console.ReadKey(true);// lấy thông tin phím bấm
- ConsoleKey.LeftArrow (RightArrow, UpArrow, DownArrow): phím ← (→, ↑, ↓)
- Console.WindowWidth: chiều rộng của cửa sổ
- Console.WindowHeight: chiều cao của cửa sổ

*Sử dụng vòng lặp: while(true){...}*

**Bài 2.4.** Định nghĩa phân số gồm có:

- Các thuộc tính: tử số, mẫu số
- Các phương thức:
  - Thiết lập không tham số (tử số là 0, mẫu số là 1), có tham số, sao chép.
  - Cộng, Trừ, Nhân, Chia hai phân số.
  - Rút gọn phân số về phân số tối giản
  - Xuất phân số dạng tử số/mẫu số

*Viết chương trình cài đặt và sử dụng lớp phân số với các yêu cầu:*

- Tạo hai phân số dùng các phương thức thiết lập;
- Thực hiện Cộng/Trừ/Nhân/Chia hai phân số;
- Xuất các kết quả rút gọn phân số sau Cộng/Trừ/Nhân/Chia.

**Bài 2.5\*.** Định nghĩa lớp Point (điểm) gồm có:

- 2 thành phần dữ liệu là tọa độ x và y.
- Các phương thức:
  - Thiết lập/khởi tạo không tham số gán tọa độ x=0, y=0;
  - Khởi tạo có tham số gán x, y bằng giá trị tham số truyền vào;
  - In ra màn hình thông tin tọa độ điểm theo định dạng (x,y)
  - Tính khoảng cách giữa 2 điểm.

*Sử dụng lớp Point thực hiện các yêu cầu:*

- Tạo một điểm dùng phương thức thiết lập không tham số.
- Tạo một điểm dùng phương thức thiết lập có tham số.
- In ra màn hình thông tin tọa độ 2 điểm và khoảng cách giữa 2 điểm.

**Bài 2.6\*** Xây dựng lớp Vé máy bay gồm:

- Các thành phần dữ liệu (Data): số hiệu chuyến bay, tên chuyến bay, ngày bay, giá vé, có chỗ ngồi hay không (bool).
- Các phương thức:
  - Khởi tạo không tham số gán số hiệu chuyến bay là “VN123”, tên chuyến bay là “Nha Trang – Hà Nội”, ngày bay là ngày hiện tại trong hệ thống, giá vé là 500, không chọn chỗ ngồi;
  - Khởi tạo có các tham số: tên chuyến bay, ngày bay, có chọn chỗ ngồi hay không; các thành phần dữ liệu còn lại gán giá trị như khởi tạo không tham số;

- Nhập thông tin cho một vé máy bay;
- Tính tiền khách phải trả: giá vé + Thuế Phí (350) + Phí chọn chỗ ngồi (50);
- Xuất thông tin vé máy bay: tên chuyến bay, ngày bay, tiền khách phải trả.

*Sử dụng lớp Vé máy bay thực hiện các yêu cầu:*

- Tạo ra 2 đối tượng Vé máy bay dùng 2 phương thức khởi tạo khác nhau;
- Xuất ra thông tin của 2 đối tượng vừa nhập;
- Nhập thông tin cho đối tượng được tạo ra bằng phương thức khởi tạo không tham số;
- Xuất ra thông tin của đối tượng có giá vé cao nhất trong 2 đối tượng.

**Bài 2.7\*.** Định nghĩa lớp Adult (người trưởng thành) gồm

- Các thành phần dữ liệu: họ tên, cân nặng, chiều cao.
- Các phương thức:
  - Khởi tạo không tham số, khởi tạo có tham số, khởi tạo sao chép.
  - Xuất ra màn hình thông tin người và tình trạng sức khỏe theo chỉ số BMI.
  - Tính số cân cần tăng/giảm để người chưa có sức khỏe tốt thành tốt, giả thiết chiều cao không thay đổi.

*Sử dụng lớp Adult để thực hiện các yêu cầu:*

- Nhập thông tin 1 người (dùng phương thức khởi tạo)
- Xuất thông tin người vừa nhập và tình trạng sức khỏe người đó ra màn hình.
- Nếu người đó chưa có sức khỏe tốt, xuất ra lời khuyên cần tăng/giảm bao nhiêu cân để có sức khỏe tốt.

*Ghi chú:* Sử dụng một số thuộc tính, phương thức của Console để nhập/xuất tiếng Việt và in ra chữ màu.

- Console.InputEncoding = Encoding.Unicode;
- Console.OutputEncoding = Encoding.Unicode;
- Console.BackgroundColor = ConsoleColor.DarkBlue;// màu nền
- Console.ForegroundColor= ConsoleColor.White;// màu chữ
- Console.Clear();// xóa màn hình

Chỉ số BMI (Body Mass Index) là chỉ số mỡ cơ thể, được tính toán dựa trên trọng lượng tính bằng kilogam và chiều cao tính bằng mét, công thức tính:  $BMI = \text{cân nặng} / (\text{chiều cao})^2$ . Với

- $BMI < 18.5$  : thiếu cân
- $18.5 \leq BMI \leq 25$ : bình thường (sức khỏe tốt)

-  $25 < BMI < 30$ : thừa cân

-  $BMI \geq 30$ : béo phì

**Bài 2.8\*.** Cho định nghĩa lớp PhepToan như dưới đây

```
class PhepToan
{
    //khai báo các thành phần dữ liệu
    float a, b;
    char c;
    //phương thức thiết lập có tham số có giá trị mặc định
    public PhepToan (float a1=0, float b1=0, char c1='+')
    {
        a = a1; b = b1;
        c = c1;
    }
    public void CapNhat()
    {
        Console.Write("nhap 2 so thuc:");
        a = float.Parse(Console.ReadLine());
        b = float.Parse(Console.ReadLine());
        Console.Write("nhap phep toan (+,-,*,/):");
        c = char.Parse(Console.ReadLine());
    }

    //tính toán và trả về giá trị
    float Tinhtoan()
    {
        float kq = 0;
        switch (c)
        {
            case '+': kq = a + b; break;
            case '-': kq = a - b; break;
            case '*': kq = a * b; break;
            case '/': kq = a / b; break;
        }
        return kq;
    }
    //xuất ra màn hình kết quả
    public void Xuat()
    {
        Console.WriteLine($"{a}{c}{b}={Tinhtoan()}");
    }
}
```

Hãy cho biết kết quả chạy các đoạn lệnh sau:

- a) PhepToan pt = new PhepToan();  
pt.Xuat();
- b) PhepToan pt = new PhepToan();  
pt.CapNhat();//giả sử nhập a=7, b=8, c=+  
pt.Xuat();
- c) PhepToan pt = new PhepToan(3,6,'\*');  
pt1.Xuat();
- d) PhepToan pt = new PhepToan(3,6,'\*');  
pt.CapNhat();//giả sử nhập a=7, b=8, c=+  
pt.Xuat();

----- *hết* -----