

# Tóm tắt kiến thức hôm nay

- ❖ Xuất ra màn hình:

`Console.WriteLine(chuỗi cần xuất);`

- ❖ Các kiểu dữ liệu: `int`, `float`, `double`,... tương tự trong C/C++, ngoài ra để thao tác với chuỗi thì dùng `string`

- ❖ Các toán tử (tương tự C/C++) :

`+` `-` `*` `/` `%` : cộng, trừ, nhân, chia nguyên, chia dư

tăng 1 đơn vị: `++`

giảm 1 đơn vị: `--`

So sánh khác: `!=`

So sánh bằng: `==`

- ❖ Nhập dữ liệu (bản chất là nhập chuỗi rồi ép về kiểu mong muốn):

Nhập chuỗi: `string a = Console.ReadLine();`

+Cú pháp ép kiểu : `kieudulieu.Parse(chuỗi giá trị cần chuyển);`

Nhập kiểu nguyên: `int x = int.Parse(Console.ReadLine());`

Nhập kiểu thực : `float f = float.Parse(Console.ReadLine());`

..các kiểu khác tương tự , duy chỉ có ép kiểu về chuỗi là sử dụng phương thức `ToString()`

Câu lệnh điều khiển : `if..else`, `switch..case` (giống hệt trong C/C++)

- ❖ Vòng lặp : (có 4 vòng lặp hay sử dụng `while`, `do while`, `for` ) - giống hệt C++

Tuy chỉ có thêm vòng lặp `foreach` (dùng để duyệt từ đầu đến cuối từng phần tử trong 1 tập hợp nào đó)

Ví dụ về xuất từng phần tử trong mảng `a`

```
int[] a = {1,2,3,7}
```

```
foreach(int temp in a)
```

```
{ Console.WriteLine(temp); }
```

## Bài tập lần 1- C# (HIT)

Bài 1: Nhập vào 5 số nguyên a, b, c, d, e. Tìm số lớn thứ 2 trong các số đó.

Bài 2. Nhập vào 3 số a, b, c (là các số thực không âm). Kiểm tra xem đó có phải là 3 cạnh của tam giác không? Nếu có thì nó thuộc loại tam giác gì (thường, cân, vuông, vuông cân hay đều).

*Chú ý: Trong 1 tam giác thì tổng 2 cạnh bất kì luôn lớn hơn hoặc bằng cạnh còn lại.*

Bài 3: Nhập vào 1 số nguyên n. Hiển thị các ước của nó ra màn hình (khi n chia hết cho m thì m đc gọi là ước của n)

Bài 4: Nhập vào số tự nhiên n. Tính S1, S2, S3 rồi in kết quả ra màn hình, biết rằng:

$$S1 = 1+2+3+...+n$$

$$S2 = 1*2*3*...*n$$

$$S3 = 1*2+2*3+3*4+...+(n-1)*n$$

Bài 5: Viết chương trình nhập vào 1 số nguyên n. Kiểm tra xem n có phải là số nguyên tố hay không ?

Bài 6\*: Nhập vào 1 số nguyên n, phân tích số đó thành tích của các thừa số nguyên tố.

VD:  $28 = 2 \times 2 \times 7$

Bài 7\*: Xây dựng công thức tính  $e^x$  với độ chính xác c (ví dụ 0.001 sẽ tính chính xác đến 3 số sau phần thập phân).

Input: Nhập vào một số nguyên x và số thực c.

Output: Tính  $e^x = 1 + \frac{x}{1!} + \frac{x^2}{2!} + \frac{x^3}{3!} + ... + \frac{x^n}{n!}$  tổng đó được xác định khi thỏa mãn  $|\frac{x^n}{n!}| < c$

