

BÀI THỰC HÀNH SỐ 4: HÀM TRONG C#.NET

I. Mục tiêu bài học

- Kiến thức:

- Củng cố kiến thức, cú pháp làm việc Hàm: Cách thức tạo Hàm, lời gọi hàm, sử dụng dạng tham số dạng tham chiếu và tham trị

- Kỹ năng

- Vận dụng làm được bài tập cơ bản về Hàm trên Window Form và Console

- Thái độ

- Khả năng tư duy, cẩn thận trong công việc

II. Bài tập có gợi ý (60p)

1. Console Application: Khai báo mảng, khởi gán giá trị ban đầu của mảng

```
static void Main(string[] args){
    string output = "";
    int[] x;           // Khai báo mảng
    x = new int[10];   // Khởi tạo mảng, thiết lập giá trị mặc định

    int[] y = { 32, 27, 64, 18, 95, 14, 90, 70, 60, 37 };

    const int ARRAY_SIZE = 10; // Hằng kích thước mảng
    int[] z;

    // Khởi tạo mảng, kích thước mảng là giá trị hằng số
    z = new int[ARRAY_SIZE];

    // Thiết lập giá trị mảng
    for (int i = 0; i < z.Length; i++)
        z[i] = 2 + 2 * i;

    output += "STT\t Mảng x\t Mảng y\t Mảng z\n";

    // Giá trị của từng mảng
    for (int i = 0; i < ARRAY_SIZE; i++)
        output += i + "\t" + x[i] + "\t" + y[i] +
            "\t" + z[i] + "\n";

    Console.WriteLine(output);
    Console.ReadLine();
}
```

2. Console Application: Mảng là tham số của hàm

// Phương thức sửa đổi các phần tử của mảng

```
public void ModifyArray( int[] b ) {
```

```
    for ( int j = 0; j < b.Length; j++ )
        b[j] *= 2;
```

```
}
```

// Phương thức sửa đổi 1 phần tử

```
public void ModifyElement (int e) {
```

```
    string ketqua = "";
```

```
    ketqua += "\nGiá trị phần tử trước khi sửa: " + e;
```

```
    e *= 2;
```

```
    ketqua += "\nGiá trị phần tử sau khi sửa: " + e;
```

```
}
```

```

static void Main(string[] args) {
    int[] a = { 1, 2, 3, 4, 5 };

    Console.WriteLine("Hiệu quả của truyền tham số là mảng " +
        "Gọi dạng tham chiếu:\n\n Giá trị phần tử mảng ban đầu");

    for (int i = 0; i < a.Length; i++)
        Console.WriteLine("    " + a[i]);
    ModifyArray(a);    // Mảng được truyền dạng tham chiếu
    Console.WriteLine("\n\nGiá trị của mảng bị thay đổi:");
    // Hiển thị các phần tử của mảng
    for (int i = 0; i < a.Length; i++)
        Console.WriteLine("    " + a[i]);
    Console.WriteLine("Sửa đổi phần tử trong mảng: " + a[3]);
    ModifyElement(a[3]);
    Console.WriteLine("a[3] thực tế sau khi bị sửa đổi: " + a[3]);
    Console.ReadLine();
}

```

3. Windows Form Application: Tạo ứng dụng tính số ngày từ thời gian nhập trong Textbox

```

public partial class Form1 : Form {
    private void btnTinh_Click(object sender, EventArgs e) {
        Ngaythang mydate = new Ngaythang();

        mydate.ngay = Int32.Parse(txtNgay.Text);
        mydate.thang = Int32.Parse(txtThang.Text);
        mydate.nam = Int32.Parse(txtNam.Text);

        lblResult.Text = "So ngay trong nam = " + mydate.ngayTrongNam();
    }
}
class Ngaythang {
    public int nam;
    public int thang;
    public int ngay;
    public static bool namNhuon(int _nam) {
        return ((_nam % 4 == 0 && _nam % 100 != 0) || (_nam % 400 == 0));
    }
    public int ngayTrongNam() {
        //So ngay tích luy trong 12 thang
    }
}

```

```

    int[] thangs = new int[] { 0, 31, 59, 90, 120, 151,
                                181, 212, 243, 273, 304, 334 };
    return thangs[thang - 1] + ngay +
           (thang > 2 && namNhuan(nam) ? 1 : 0);
}
}

```

III. Bài tập tự làm

1. Tạo mảng là biến toàn cục và hàm thực hiện tìm kiếm 1 phần tử trong mảng với giá trị tìm kiếm nhập từ bàn phím

2. Sử dụng thuật toán sắp xếp nổi bọt, sắp xếp mảng theo thứ tự tăng dần

```

public static void BubbleSort(int[] b){
    for (int j = 1; j < b.Length; j++){
        for (int i = 0; i < b.Length - 1; i++){
            if (b[i] > b[i + 1])
                Swap(b, i);           // Hoán đổi
        }

        //Hoán đổi 2 phần tử của mảng
    }
}

public static void Swap(int[] c, int first){
    int hold;           // Biến tạm lưu trữ phần tử hoán đổi
    hold = c[first];
    c[first] = c[first + 1];
    c[first + 1] = hold;
}

```