

## Nội dung tuần 01

Luyện tập cách xây dựng Hàm dựng và Hàm hủy cho lớp đối tượng.

### Bài tập

❖ Yêu cầu

A: 2 bài bất kỳ

H: Làm hết 5 bài

Lưu ý: mục tiêu quan trọng là các thành phần dữ liệu hợp lệ của các lớp đối tượng, gợi ý kết quả chỉ là một trong số các cách khởi tạo giá trị với tham số.

Khi khởi tạo với nhiều thành phần dữ liệu có mối liên quan với nhau thì nên thực hiện mối liên quan đó (ví dụ số ngày dư thì đổi sang tháng, tháng dư thì đổi sang năm,...)

### Bài 1

Cài đặt lại ví dụ về các phương thức khởi tạo và hủy được gọi.

### Bài 2

Khai báo và cài đặt lớp Ngày sao cho hàm main sau chạy đúng.

```
int main()
{
    Ngày n1;                /// 1/1/1
    n1.Xuat();
    cout << endl;
    Ngày n2(2, 10, 2024);   /// 2/10/2024
    n2.Xuat();
    cout << endl;
    Ngày n3(-100, 16, 1990); /// 10/7/1991
    n3.Xuat();
    cout << endl;
    Ngày n4(1403);          /// 23/11/4
    n4.Xuat();
    cout << endl;
    Ngày n5("12-12-2012");  /// 12/12/2012
    n5.Xuat();
    cout << endl << endl;

    system("pause");
}
```

### Bài 3

Khai báo và cài đặt lớp Thời Gian để chạy đúng với hàm main sau:

```
void main()
{
    ThoiGian tg1;                //00:00:00
    tg1.Xuat();
    ThoiGian tg2(1212);          //00:20:12
    tg2.Xuat();
    ThoiGian tg3(125,45);        //02:05:45
    tg3.Xuat();
    ThoiGian tg4(12,239,-78);    //15:59:00
    tg4.Xuat();
}
```

### Bài 4

Khai báo và cài đặt lớp Sinh Viên để chạy đúng với hàm main sau

```
void main()
{
    //MSSV: "xxxxxxx", HoTen: "xxxx", NgaySinh: xx/xx/xxxx, DLT: 10, DTH: 10.
    SinhVien sv1;
    sv1.Xuat();
    SinhVien sv2("1363001");
    sv2.Xuat();
    SinhVien sv3("1363002", "Nguyen Van A", 7, 8.5);
    sv3.Xuat();
    SinhVien sv4("1363003", "Tran Thi B", 8, 9.5, 12, 3, 1996);
    sv4.Xuat();
    NgayThang n(30,2,1996);
    SinhVien sv5("1363004", "Ngo Van C", 5, 6, n);
    sv5.Xuat();
    SinhVien sv6(sv5);
    sv6.Xuat();
}
```

### Bài 5

Khai báo và cài đặt lớp Mảng Số Nguyên để chạy đúng với hàm main sau

```
void main()
{
    MangSoNguyen msn1;
    msn1.Xuat();
    MangSoNguyen msn2(5);
    msn2.Xuat();
    int arr[10];
    for (int i = 0; i < 8; i++)
    {
        arr[i] = i * i;
    }
    MangSoNguyen msn3(arr, 8);
    msn3.Xuat();
    MangSoNguyen msn4(msn3);
    msn4.Xuat();
}
```

### Hướng dẫn

#### Constructor & Destructor

Xem xét thứ tự khởi tạo cũng như hủy đối tượng (thời điểm hàm dựng hay hàm hủy được gọi) thông qua ví dụ sau:

```
static int numMax = 0;
class Object
{
private:
    int info;
    int curNum;

public:
    Object();
    Object(int value);
    ~Object();
};

Object::Object()
```

```
{
    info = 0;
    curNum = ++numMax;
    cout << "Khoi tao (mac dinh) doi tuong " << curNum << endl;
}

Object::Object(int value)
{
    info = value;
    curNum = ++numMax;
    cout << "Khoi tao (co doi so) doi tuong " << curNum << endl;
}

Object::~~Object()
{
    cout << "Huy doi tuong " << curNum << endl;
}

void Create()
{
    Object obj1(5);    //biến cục bộ hàm Create
    Object obj2;        //biến cục bộ hàm Create
}

Object objGlobal;    //biến toàn cục
void main()
{
    Object obj1;        //biến cục bộ hàm main
    Create();            //gọi hàm Create()
    Object obj2(3);    //biến cục bộ hàm main
}
```

Có nhận xét gì về kết quả xuất của chương trình.

```
Khoi tao (mac dinh) doi tuong 1
Khoi tao (mac dinh) doi tuong 2
Khoi tao (co doi so) doi tuong 3
Khoi tao (mac dinh) doi tuong 4
Huy doi tuong 4
Huy doi tuong 3
Khoi tao (co doi so) doi tuong 5
Huy doi tuong 5
Huy doi tuong 2
Huy doi tuong 1
Press any key to continue . . .
```

### Class Ngay

Xét ở mức đơn giản mỗi tháng đều có 30 ngày.

```
class Ngay
{
private:
    int _ng, _th, _n;
    void ChuanHoa();
public:
```

```
    Ngay();
    Ngay(const int& );
    Ngay(const int&, const int&, const int& );
    Ngay(const char* s);
    ~Ngay();
    void Xuat();
};

void Ngay::ChuanHoa()
{
    _ng = abs(_ng);
    _th = abs(_th);
    _n = abs(_n);
    int thangDu = (_ng - 1) / 30;
    _ng = (_ng - 1) % 30 + 1;
    _th += thangDu;
    int namDu = (_th - 1) / 12;
    _th = (_th - 1) % 12 + 1;
    _n += namDu;
}

Ngay::Ngay()
{
    _ng = _th = _n = 1;
}

Ngay::Ngay(const int& x)
{
    _ng = abs(x);
    _th = _n = 1;
    ChuanHoa();
}

Ngay::Ngay(const int& ng, const int& th, const int& n)
{
    _ng = abs(ng);
    _th = abs(th);
    _n = abs(n);
    ChuanHoa();
}

Ngay::Ngay(const char* s)
{
    int len = strlen(s);
    char* sTemp = new char[len + 1];
    strcpy_s(sTemp, len + 1, s);
    const char* delimiters = "-/";
    char* temp;
    char* sCur = strtok_s(sTemp, delimiters, &temp);
    _ng = atoi(sCur);
    if (sCur)
    {
        sCur = strtok_s(nullptr, delimiters, &temp);
        _th = atoi(sCur);
    }
    if (sCur)
    {
        sCur = strtok_s(nullptr, delimiters, &temp);
```

```
        _n = atoi(sCur);
    }
    delete[] sTemp;
    ChuanHoa();
}

Ngay::~Ngay()
{
}

void Ngay::Xuat()
{
    cout << _ng << "/" << _th << "/" << _n;
}
```