

BÀI TẬP THỰC HÀNH TUẦN 2

- Làm các bài tập sau dưới dạng hướng đối tượng. Ngôn ngữ C++
- Bài làm trên một Solution đặt tên là "MSSV". Project đặt tên là "Tuan02"
- Sau khi làm bài xong nén lại thành một file .rar duy nhất đặt tên <MSSV>.rar sau đó nộp lên moodle.
- *Các bài đặt tên sai quy định và sinh viên chép bài nhau đều bị 0 điểm.*

Cài đặt lớp **CIntArray** viết chương trình theo các yêu cầu sau:

- Attributes (Các thuộc tính)
 - o m_Array (mảng chứa các phần tử số nguyên)
 - o m_Length (số phần tử trong mảng)
- Constructors (Phương thức khởi tạo)
 - o Default constructor (Phương thức khởi tạo mặc định)
 - o Copy constructor (Phương thức khởi tạo sao chép)
 - o Constructor with paramater (Phương thức khởi tạo với tham số đầu vào)
- Methods (Các phương thức cài đặt)

Tên phương thức và kiểu trả về	Ghi chú
InputArray: void	Phương thức nhập (nhập tự động)
OutputArray: void	Phương thức xuất
Remove(): CIntArray	Xóa các phần tử trùng trong mảng
Remove (int): CIntArray	Xóa bỏ phần tử tại vị trí bất kỳ
Remove (int, int): CIntArray	Xóa bỏ n phần tử, tại vị trí bắt đầu từ phần tử ở vị trí k nào đó.
Replace (int, int): CIntArray	Thay tất cả phần tử có giá trị cũ thành giá trị mới
AddHead (int): CIntArray	Thêm một phần tử vào đầu mảng
AddTail (int): CIntArray	Thêm một phần tử vào cuối mảng
Insert (int, int): CIntArray	Thêm một phần tử vào vị trí k nào đó
Max(): int	Tìm phần tử lớn nhất của mảng
Min(): int	Tìm phần tử nhỏ nhất của mảng
Ascending(): CIntArray	Sắp xếp mảng tăng dần
Descending: CIntArray	Sắp xếp mảng giảm dần
IsSymmetry(): bool	Kiểm tra mảng có đối xứng hay không
MoveForward(): CIntArray	Dịch chuyển mảng về phía trước 1 đơn vị
MoveForward(int): CIntArray	Dịch chuyển mảng về phía trước n đơn vị
MoveBehind(): CIntArray	Dịch chuyển mảng về phía sau 1 đơn vị
MoveBehind(int): CIntArray	Dịch chuyển mảng về phía sau n đơn vị
Invert(): CIntArray	Đảo ngược mảng ban đầu
Appearance(int): int	Tìm tần suất xuất hiện của một phần tử trong mảng

Hướng dẫn:

Bước 0: Tạo một project rỗng Chọn loại dự án là **C++ / Console Application**.

Bước 1: Tạo lớp với các thuộc tính như trong đề bài yêu cầu.

Bước 2: Cài đặt trước 3 loại phương thức khởi tạo cần làm như sau, trong ví dụ bên dưới là của lớp Điểm, lớp trong đề bài dựa vào nội dung bên dưới để viết lại:

Default constructor (Phương thức khởi tạo mặc định)

```
class CDiem
{
private:
    float x;
    float y;
public:
    void Nhap();
    void Xuat();
    float KhoangCach(CDiem B);
    CDiem(void);
    ~CDiem(void);
};
```

```
CDiem::CDiem(void)
{
    x = 0;
    y = 0;
}
```

Copy constructor (Phương thức khởi tạo sao chép)

```
class CDiem
{
private:
    float x;
    float y;
public:
    void Nhap();
    void Xuat();
    const int* Test();
    int Test2();
    const float& KhoangCach(CDiem B);
    CDiem(void);
    CDiem(float z);
    CDiem(const CDiem &D);
    ~CDiem(void);
};
```

```
CDiem::CDiem(const CDiem &D)
{
    x = D.x;
    y = D.y;
}
```

Constructor with paramater (Phương thức khởi tạo với tham số đầu vào)

```
class CDiem
{
private:
    float x;
    float y;
public:
    void Nhap();
    void Xuat();
    float KhoangCach(CDiem B);
    CDiem(void);
    CDiem(float z);
    ~CDiem(void);
};
```

```
CDiem::CDiem(float z)
{
    x = z;
    y = z;
}
```

Bước 3: Khai báo và cài đặt các phương thức còn lại trong đề bài và sau đó định nghĩa trong file *.cpp.

Bước 4: Ở mỗi phương thức gọi lại trong hàm main để kiểm tra kết quả.

HẾT