

**ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH**  
**TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN**  
**KHOA CÔNG NGHỆ PHẦN MỀM**

---



**MÔN LẬP TRÌNH HƯỚNG ĐỐI TƯỢNG**  
**BÀI TẬP THỰC HÀNH 3**

GVHD: Nguyễn Ngọc Quý

Sinh viên thực hiện: Khương Thành Lên - 24520949

## NHẬN XÉT CỦA GIÁO VIÊN HƯỚNG DẪN

[illegible]

....., ngày.....tháng.....năm 2025

**Người nhận xét**

(Ký tên và ghi rõ họ tên)

## MỤC LỤC

<b>Bài tập 1.....</b>	<b>7</b>
<b>Bài tập 2.....</b>	<b>16</b>
<b>Bài tập 3.....</b>	<b>25</b>
<b>Bài tập 4.....</b>	<b>33</b>
<b>Bài tập 5.....</b>	<b>38</b>
<b>Bài tập 6.....</b>	<b>44</b>
<b>Bài tập 7.....</b>	<b>54</b>
<b>Bài tập 8.....</b>	<b>62</b>
<b>LINK: <a href="https://github.com/ThanhLen-IT/IT002-Lab03">https://github.com/ThanhLen-IT/IT002-Lab03</a> .....</b>	<b>70</b>

**DANH MỤC HÌNH ẢNH**

Hình 1. Class diagram Bài 1 .....	7
Hình 2. Kiểm thử 1 (cTamGiac).....	13
Hình 3. Kiểm thử 2 (cTamGiac).....	13
Hình 4. Kiểm thử 3 (cTamGiac).....	14
Hình 5. Kiểm thử 4 (cTamGiac).....	14
Hình 6. Kiểm thử 5 (cTamGiac).....	15
Hình 7. Kiểm thử 6 (cTamGiac).....	15
Hình 8. Kiểm thử 7 (cTamGiac).....	16
Hình 9. Kiểm thử 8 (cTamGiac).....	16
Hình 10. Class diagram Bài 2 .....	17
Hình 11. Kiểm thử 1 (cDaGiac) .....	22
Hình 12. Kiểm thử 2 (cDaGiac) .....	23
Hình 13. Kiểm thử 3 (cDaGiac) .....	23
Hình 14. Kiểm thử 4 (cDaGiac) .....	23
Hình 15. Kiểm thử 5 (cDaGiac) .....	24
Hình 16. Kiểm thử 6 (cDaGiac) .....	24
Hình 17. Kiểm thử 7 (cDaGiac) .....	25
Hình 18. Kiểm thử 8 (cDaGiac) .....	25
Hình 19. Class diagram Bài 3 .....	26
Hình 20. Kiểm thử 1 (cDSPhanSo) .....	31
Hình 21. Kiểm thử 2 (cDSPhanSo) .....	32
Hình 22. Kiểm thử 3 (cDSPhanSo) .....	32
Hình 23. Kiểm thử 4 (cDSPhanSo) .....	33
Hình 24. Kiểm thử 5 (cDSPhanSo) .....	33
Hình 25. Calss digaram Bài 4.....	34
Hình 26. Kiểm thử 1 (cArray) .....	37
Hình 27. Kiểm thử 2 (cArray) .....	38
Hình 28. Kiểm thử 3 (cArray) .....	38
Hình 29. Class digram Bài 5.....	39
Hình 30. Kiểm thử 1 (cDaThuc).....	42
Hình 31. Kiểm thử 2 (cDaThuc).....	43
Hình 32. Kiểm thử 3 (cDaThuc).....	43
Hình 33. Kiểm thử 4 (cDaThuc).....	44
Hình 34. Kiểm thử 5 (cDaThuc).....	44
Hình 35. Class diagram Bài 6.....	45
Hình 36. Kiểm thử 1 (thí sinh) .....	50
Hình 37. Kiểm thử 2 (thí sinh) .....	50
Hình 38. Kiểm thử 3 (thí sinh) .....	51
Hình 39. Kiểm thử 4 (thí sinh) .....	51
Hình 40. Kiểm thử 5 (thí sinh) .....	52
Hình 41. Kiểm thử 6 (thí sinh) .....	52
Hình 42. Kiểm thử 7 (thí sinh) .....	53
Hình 43. Kiểm thử 8 (thí sinh) .....	54

## IT002 – Lập trình hướng đối tượng

Hình 44. Class diagram Bài 7 .....	55
Hình 45. Kiểm thử 1 (NhanvienVP).....	60
Hình 46. Kiểm thử 2 (NhanvienVP).....	60
Hình 47. Kiểm thử 3 (NhanvienVP).....	61
Hình 48. Kiểm thử 4 (NhanvienVP).....	61
Hình 49. Kiểm thử 5 (NhanvienVP).....	61
Hình 50. Kiểm thử 6 (NhanvienVP).....	62
Hình 51. Class diagram Bài 8 .....	62
Hình 52. Kiểm thử 1 (NhanVienSX).....	68
Hình 53. Kiểm thử 2 (NhanVienSX).....	68
Hình 54. Kiểm thử 3 (NhanVienSX).....	69
Hình 55. . Kiểm thử 4 (NhanVienSX).....	69
Hình 56. Kiểm thử 5 (NhanVienSX).....	69
Hình 57. Kiểm thử 6 (NhanVienSX).....	70
Hình 58. Kiểm thử 7 (NhanVienSX).....	70

## DANH MỤC BẢNG

Bảng 1. cDiem.h .....	7
Bảng 2. cDiem.cpp .....	9
Bảng 3. cTamGiac.h .....	9
Bảng 4. cTamGiac.cpp .....	11
Bảng 5. main.cpp (cTamGiac).....	11
Bảng 6. Test cases cTamGiac.....	12
Bảng 7. cDiem.h .....	17
Bảng 8. cDiem.cpp .....	18
Bảng 9. cDaGiac.h.....	19
Bảng 10. cDaGiac.cpp .....	20
Bảng 11. main.cpp (cDaGiac) .....	21
Bảng 12. Test cases cDaGiac .....	22
Bảng 13. cPhanSo.h.....	26
Bảng 14. cPhanSo.cpp .....	28
Bảng 15. cDSPhanSo.h.....	28
Bảng 16. cDSPhanSo.cpp .....	30
Bảng 17. main.cpp (cDSPhanSo) .....	30
Bảng 18. Test cases cDSPhanSO .....	31
Bảng 19. cArray.h.....	34
Bảng 20. cArray.cpp.....	36
Bảng 21. main.cpp (cArray) .....	37
Bảng 22. Test cases cArray .....	37
Bảng 24. cDaThuc.h .....	39
Bảng 25. cDaThuc.cpp .....	41
Bảng 26. main.cpp(cDaThuc).....	41
Bảng 27. Test cases cDaThuc.....	42
Bảng 28. cCandidate.h.....	45

## IT002 – Lập trình hướng đối tượng

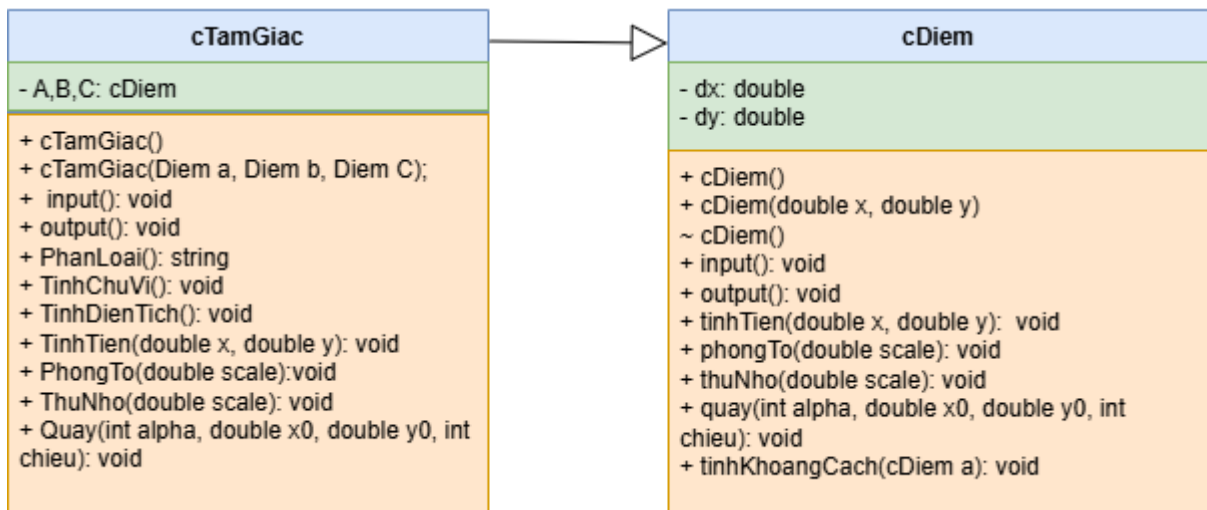
Bảng 29. cCandidate.cpp .....	46
Bảng 30. cListCandidate.h .....	47
Bảng 31. cCandidate.cpp .....	48
Bảng 32. mani.cpp (thí sinh) .....	49
Bảng 33. Test cases thí sinh .....	49
Bảng 34. cNhanVienVP.h .....	56
Bảng 35. cNhanVienVP.cpp.....	57
Bảng 36. cListNhanVienVP.h .....	57
Bảng 37. cListNhanVienVP.cpp .....	58
Bảng 38. main.cpp (NhanVienVP).....	59
Bảng 39. Test cases NhanVienVP .....	59
Bảng 40. cNhanVienSX.h .....	63
Bảng 41. cNhanVienSX.cpp.....	64
Bảng 42. cListNhanVienSX.h .....	65
Bảng 43. cListNhanVienSX.cpp .....	66
Bảng 44. main.cpp (NhanVienSX).....	67
Bảng 45. Test cases NhanVienSX.....	67

## NỘI DUNG BÀI LÀM

### Bài tập 1

Định nghĩa lớp **cTamGiac** để biểu diễn khái niệm tam giác trong mặt phẳng (gồm 3 đỉnh A, B, C) với các phương thức thiết lập, hủy bỏ (nếu có). Các hàm thành phần: thiết lập, nhập, xuất, kiểm tra loại tam giác (thường, vuông, cân, đều), tính chu vi, tính diện tích, tịnh tiến, quay, phóng to, thu nhỏ, *vẽ tam giác (nếu được)*. Viết chương trình minh họa các chức năng của lớp cTamGiac. *Lưu ý: Để vẽ hình sinh viên cần tìm hiểu thư viện đồ họa (graphic), yêu cầu này tùy chọn không bắt buộc.*

Nội dung diagram của chương trình lớp cTamGiac:



Hình 1. Class diagram Bài 1

Nội dung code của chương trình lớp cTamGiac:

```

#pragma once
#include <iostream>

class cDiem
{
private:
    double dx;
    double dy;
public:
    cDiem() {
        this->dx = 0;
        this->dy = 0;
    }
    cDiem(float x, float y) {
        this->dx = x;
        this->dy = y;
    }
    void input();
    void output();
    void tinhTien(double x, double y);
    void quay(int alpha, double x, double y, int chieu);
    void phongTo(float);
    void thuNho(float);
    double tinhKhoangCach(cDiem a);
};
  
```

Bảng 1. cDiem.h

```

#include "cDiem.h"
#include <cmath>
using namespace std;

//Định nghĩa số pi
#ifndef PI
#define PI 3.14159265358979323846
#endif

void cDiem::setDiem(double x, double y) {
    this->dx = x;
    this->dy = y;
}

void cDiem::input() {
    cin >> dx >> dy;
}

void cDiem::output() {
    cout << "(" << dx << "," << dy << ")" << " ";
}

// Tinh tien diem
// Input: 2 so thuc x, y; Output: diem sau khi tinh tien
void cDiem::tinhTien(double x, double y) {
    dx += x;
    dy += y;
}

// Quay diem quanh diem (x0, y0)
// Input: 3 so nguyen alpha, x0, y0, chieu(1: chieu am; 2: chieu duong );
// Output: diem sau khi quay
void cDiem::quay(int alpha, double x0, double y0, int chieu) {
    double rad = alpha * PI / 180;
    double x = dx - x0;
    double y = dy - y0;

    double dx_new, dy_new;

    if (chieu == 1) { // xoay cùng chiều kim đồng hồ(chiều âm)
        dx_new = x0 + x * cos(rad) + y * sin(rad);
        dy_new = y0 - x * sin(rad) + y * cos(rad);
    }
    else { // ngược chiều
        dx_new = x0 + x * cos(rad) - y * sin(rad);
        dy_new = y0 + x * sin(rad) + y * cos(rad);
    }

    dx = dx_new;
    dy = dy_new;
}

// Phóng to điểm
// Input: 1 so thuc scale; Output: diem sau khi phong to
void cDiem::phongTo(float scale) {
    dx *= scale;
    dy *= scale;
}

// Thu nhỏ điểm
// Input: 1 so thuc scale; Output: diem sau khi thu nho
void cDiem::thuNho(float scale) {
    dx /= scale;
    dy /= scale;
}

```



```
// Tính khoảng cách giữa 2 điểm
// Input: 1 điểm a; Output: khoảng cách giữa 2 điểm
double cDiem::tinhKhoangCach(cDiem a) {
    return sqrt(pow(dx - a.dx, 2) + pow(dy - a.dy, 2));
}
```

Bảng 2. cDiem.cpp

```
#pragma once
#include "cDiem.h"
#include <string>

using namespace std;
class cTamGiac : public cDiem
{
private:
    cDiem A, B, C;
public:
    cTamGiac() : A(0, 0), B(0, 0), C(0, 0) {};
    cTamGiac(cDiem a, cDiem b, cDiem c) {
        A = a;
        B = b;
        C = c;
    }
    ~cTamGiac() {
        return;
    };
    void input();
    void output();
    string PhanLoai();
    double TinhChuVi();
    double TinhDienTich();
    void TinhTien(double x, double y);
    void PhongTo(double scale);
    void ThuNho(double scale);
    void Quay(int alpha, float x0, float y0, int chieu);
};
```

Bảng 3. cTamGiac.h

```
#include "cTamGiac.h"
#include <cmath>

void cTamGiac::input() {
    cout << "Nhap diem A: ";
    A.input();
    cout << "Nhap diem B: ";
    B.input();
    cout << "Nhap diem C: ";
    C.input();
}

void cTamGiac::output() {
    cout << "A"; A.output();
    cout << " B"; B.output();
    cout << " C"; C.output();
    cout << endl;
}

//Input: Không có; Output: Phân loại tam giác
//tam giác đều(2 cạnh bằng nhau) , tam giác cân(3 cạnh bằng nhau),
//tam giác vuông(pytago), tam giác thường(con lại)
string cTamGiac::PhanLoai() {
    double a = A.tinhKhoangCach(B);
```

```

        double b = B.tinhKhoangCach(C);
        double c = A.tinhKhoangCach(C);
        if (a == b && b == c) {
            return "Tam giac deu";
        }
        else if (a == b || b == c || a == c) {
            return "Tam giac can";
        }
        else if (a * a + b * b == c * c || a * a + c * c == b * b || b * b + c *
c == a * a) {
            return "Tam giac vuong";
        }
        else {
            return "Tam giac thuong";
        }
    }

//Input: Khong co; Output: Chu vi tam giac
//Thuat toan: Chu vi = do dai AB + do dai BC + do dai AC
double cTamGiac::TinhChuVi() {
    return A.tinhKhoangCach(B) + B.tinhKhoangCach(C) + A.tinhKhoangCach(C);
}

//Input: Khong co; Output: Dien tich tam giac
//Thuat toan: Dien tich = can bac hai cua (p * (p - a) * (p - b) * (p - c))
double cTamGiac::TinhDienTich() {
    double p = TinhChuVi() / 2;
    double a = A.tinhKhoangCach(B);
    double b = B.tinhKhoangCach(C);
    double c = A.tinhKhoangCach(C);
    return sqrt(p * (p - a) * (p - b) * (p - c));
}

//Input: Khong co; Output: Tam giac da tinh tien
//Thuat toan: Tinh tien tung diem A, B, C
void cTamGiac::TinhTien(double x, double y) {
    A.tinhTien(x,y);
    B.tinhTien(x,y);
    C.tinhTien(x, y);
    cout << "Tam giac sau khi tinh tien: ";
    output();
}

//Input: he so phong to; Output: Tam giac da phong to
//Thuat toan: Phong to tung diem A, B, C
void cTamGiac::PhongTo(double scale) {
    A.phongTo(scale);
    B.phongTo(scale);
    C.phongTo(scale);
    cout << "Tam giac sau khi phong to: ";
    output();
}

//Input: he so thu nho; Output: Tam giac da thu nho
//Thuat toan: Thu nho tung diem A, B, C
void cTamGiac::ThuNho(double scale) {
    A.thuNho(scale);
    B.thuNho(scale);
    C.thuNho(scale);
    cout << "Tam giac sau khi thu nho: ";
    output();
}

//Input: 3 so nguyen alpha, x0, y0, chieu(1: chieu am; 2: chieu duong );
Output: Tam giac da quay

```

## IT002 – Lập trình hướng đối tượng

```
//Thuat toan: Quay tung diem A, B, C
void cTamGiac::Quay(int alpha, float x0, float y0, int chieu) {
    A.quay(alpha, x0, y0, chieu);
    B.quay(alpha, x0, y0, chieu);
    C.quay(alpha, x0, y0, chieu);
    output();
}
```

Bảng 4. cTamGiac.cpp

```
#include "cTamGiac.h"

int main()
{
    cTamGiac tg;
    tg.input();
    tg.output();
    cout << "Loai tam giac: " << tg.PhanLoai()<<endl;
    cout << "Chu vi: " << tg.TinhChuVi()<<endl;
    cout << "Dien tich: " << tg.TinhDienTich()<<endl;
    double x, y;
    cout << "Nhap vector tinh tien: "; cin >> x >> y;
    tg.TinhTien(x, y);
    double scale;
    cout << "Nhap ti le phong to/ thu nho: "; cin >> scale;
    tg.PhongTo(scale);
    tg.ThuNho(scale);
    int alpha, chieu;
    double x0, y0;
    cout << "Nhap lan luot goc quay, tam x0, tam y0, chieu (1/2=am/duong): ";
    cin >> alpha >> x0 >> y0 >> chieu;
    tg.Quay(alpha, x0, y0, chieu);
}
```

Bảng 5. main.cpp (cTamGiac)

Nội dung kiểm thử của chương trình lớp cTamGiac:

ST T	A	B	C	Vector Tinh Tien (x, y)	Tỉ Lệ Thu/ Phóng	Góc Quay ( $\alpha$ )	Tâm Qua y (x0, y0)	Chiều (1:âm, 2: dương)	Mục Đích
1	(0,0)	(1,0)	(0.5,0.866)	(1,1)	2.0	90	(0,0)	2	Kiểm tra tịnh tiến, phóng to, quay ngược chiều
2	(0,0)	(2,0)	(1,2)	(-1,-1)	0.5	180	(0,0)	1	Kiểm tra thu nhỏ, quay

**IT002 – Lập trình hướng đối tượng**

									180 độ chiều âm
3	(0,0)	(3,0)	(0,4)	(0,0)	1.0	0	(0,0)	2	Không biến đổi, kiểm tra tam giác vuông
4	(1,1)	(2,3)	(4,2)	(-2,-2)	2.0	45	(1,1)	2	Nhiều biến đổi liên tiếp
5	(0,0)	(1,0)	(2,0)	(0,0)	1.0	0	(0,0)	1	Kiểm tra loại tam giác không hợp lệ
6	(1,1)	(1,1)	(1,1)	(0,0)	1.0	0	(0,0)	2	Ba điểm trùng nhau, kiểm tra xử lý ngoại lệ
7	(0,0)	(2,2)	(3,0)	(2,1)	0.8	60	(2,2)	2	Biến đổi nâng cao với tâm quay không gốc tọa độ
8	(1,2)	(4,6)	(7,2)	(1,-1)	0.5	270	(0,0)	1	Kiểm tra quay chiều âm, nhiều thao tác

Bảng 6. Test cases cTamGiac

Kiểm thử 1: A(0,0), B(1,0), C(0.5,0.866), vector tịnh tiến(1,1), tỉ lệ thu phóng=2.0, góc quay=90, tâm quay(0,0), chiều dương

```

Nhap diem A: 0 0
Nhap diem B: 1 0
Nhap diem C: 0.5 0.8666
A(0,0) B(1,0) C(0.5,0.8666)
Loại tam giác: Tam giác cân
Chu vi: 3.001
Diện tích: 0.4333
Nhap vector tịnh tiến: 1 1
Tam giác sau khi tịnh tiến: A(1,1) B(2,1) C(1.5,1.8666)
Nhap tỉ lệ phóng to/ thu nhỏ: 2.0
Tam giác sau khi phóng to: A(2,2) B(4,2) C(3,3.7332)
Tam giác sau khi thu nhỏ: A(1,1) B(2,1) C(1.5,1.8666)
Nhap lần lượt góc quay, tâm x0, tâm y0, chiều (1/2=âm/duong): 90 0 0 2
A(-1,1) B(-1,2) C(-1.8666,1.5)

```

Hình 2. Kiểm thử 1 (cTamGiac)

Kiểm thử 2: A(0,0), B(2,0), C(1,2), vector tịnh tiến(-1,-1), tỉ lệ thu phóng=0.5, góc quay=180, tâm quay(0,0), chiều âm

```

Nhap diem A: 0 0
Nhap diem B: 2 0
Nhap diem C: 1 2
A(0,0) B(2,0) C(1,2)
Loại tam giác: Tam giác cân
Chu vi: 6.47214
Diện tích: 2
Nhap vector tịnh tiến: -1 -1
Tam giác sau khi tịnh tiến: A(-1,-1) B(1,-1) C(0,1)
Nhap tỉ lệ phóng to/ thu nhỏ: 0.5
Tam giác sau khi phóng to: A(-0.5,-0.5) B(0.5,-0.5) C(0,0.5)
Tam giác sau khi thu nhỏ: A(-1,-1) B(1,-1) C(0,1)
Nhap lần lượt góc quay, tâm x0, tâm y0, chiều (1/2=âm/duong): 180 0 0 1
A(1,1) B(-1,1) C(1.22465e-16,-1)

```

Hình 3. Kiểm thử 2 (cTamGiac)

Kiểm thử 3: A(0,0), B(3,0), C(0,4), vector tịnh tiến(0,0), tỉ lệ thu phóng=1.0, góc quay=0, tâm quay(0,0), chiều dương

```

Nhap diem A: 0 0
Nhap diem B: 3 0
Nhap diem C: 0 4
A(0,0) B(3,0) C(0,4)
Loại tam giác: Tam giác vuông
Chu vi: 12
Diện tích: 6
Nhap vector tịnh tiến: 0 0
Tam giác sau khi tịnh tiến: A(0,0) B(3,0) C(0,4)
Nhap tỉ lệ phóng to/ thu nhỏ: 1.0
Tam giác sau khi phóng to: A(0,0) B(3,0) C(0,4)
Tam giác sau khi thu nhỏ: A(0,0) B(3,0) C(0,4)
Nhap lần lượt góc quay, tâm x0, tâm y0, chiều (1/2=âm/duong): 0 0 0 2
A(0,0) B(3,0) C(0,4)

```

Hình 4. Kiểm thử 3 (cTamGiac)

Kiểm thử 4: A(1,1), B(2,3), C(4,2), vector tịnh tiến(-2,-2), tỉ lệ thu phóng=2.0, góc quay=45, tâm quay(1,1), chiều dương

```

Nhap diem A: 1 1
Nhap diem B: 2 3
Nhap diem C: 4 2
A(1,1) B(2,3) C(4,2)
Loại tam giác: Tam giác cân
Chu vi: 7.63441
Diện tích: 2.5
Nhap vector tịnh tiến: -2 -2
Tam giác sau khi tịnh tiến: A(-1,-1) B(0,1) C(2,0)
Nhap tỉ lệ phóng to/ thu nhỏ: 2.0
Tam giác sau khi phóng to: A(-2,-2) B(0,2) C(4,0)
Tam giác sau khi thu nhỏ: A(-1,-1) B(0,1) C(2,0)
Nhap lần lượt góc quay, tâm x0, tâm y0, chiều (1/2=âm/duong): 45 1 1 2
A(1,-1.82843) B(0.292893,0.292893) C(2.41421,1)

```

Hình 5. Kiểm thử 4 (cTamGiac)

Kiểm thử 5: A(0,0), B(1,0), C(2,0), vector tịnh tiến(0,0), tỉ lệ thu phóng=1.0, góc quay=0, tâm quay(0,0), chiều âm

```

Nhap diem A: 0 0
Nhap diem B: 1 0
Nhap diem C: 2 0
A(0,0) B(1,0) C(2,0)
Loai tam giac: Tam giac can
Chu vi: 4
Dien tich: 0
Nhap vector tinh tien: 0 0
Tam giac sau khi tinh tien: A(0,0) B(1,0) C(2,0)
Nhap ti le phong to/ thu nho: 1.0
Tam giac sau khi phong to: A(0,0) B(1,0) C(2,0)
Tam giac sau khi thu nho: A(0,0) B(1,0) C(2,0)
Nhap lan luot goc quay, tam x0, tam y0, chieu (1/2=am/duong): 0 0 0 1
A(0,0) B(1,0) C(2,0)

```

Hình 6. Kiểm thử 5 (cTamGiac)

Kiểm thử 6: A(1,1), B(1,1), C(1,1), vector tịnh tiến(0,0), tỉ lệ thu phóng=1.0, góc quay=0, tâm quay(0,0), chiều dương

```

Nhap diem A: 1 1
Nhap diem B: 1 1
Nhap diem C: 1 1
A(1,1) B(1,1) C(1,1)
Loai tam giac: Tam giac deu
Chu vi: 0
Dien tich: 0
Nhap vector tinh tien: 0 0
Tam giac sau khi tinh tien: A(1,1) B(1,1) C(1,1)
Nhap ti le phong to/ thu nho: 1.0
Tam giac sau khi phong to: A(1,1) B(1,1) C(1,1)
Tam giac sau khi thu nho: A(1,1) B(1,1) C(1,1)
Nhap lan luot goc quay, tam x0, tam y0, chieu (1/2=am/duong): 0 0 0 2
A(1,1) B(1,1) C(1,1)

```

Hình 7. Kiểm thử 6 (cTamGiac)

Kiểm thử 7: A(0,0), B(2,2), C(3,0), vector tịnh tiến(2,1), tỉ lệ thu phóng=0.8, góc quay=60, tâm quay(2,2), chiều dương

```

Nhap diem A: 0 0
Nhap diem B: 2 2
Nhap diem C: 3 0
A(0,0) B(2,2) C(3,0)
Loai tam giac: Tam giac thuong
Chu vi: 8.0645
Dien tich: 3
Nhap vector tinh tien: 2 1
Tam giac sau khi tinh tien: A(2,1) B(4,3) C(5,1)
Nhap ti le phong to/ thu nho: 0.8
Tam giac sau khi phong to: A(1.6,0.8) B(3.2,2.4) C(4,0.8)
Tam giac sau khi thu nho: A(2,1) B(4,3) C(5,1)
Nhap lan luot goc quay, tam x0, tam y0, chieu (1/2=am/duong): 60 2 2 2
A(2.86603,1.5) B(2.13397,4.23205) C(4.36603,4.09808)

```

Hình 8. Kiểm thử 7 (cTamGiac)

Kiểm thử 8: A(1,2), B(4,6), C(7,2), vector tịnh tiến(1,-1), tỉ lệ thu phóng=0.5, góc quay=270, tâm quay(0,0), chiều âm

```

Nhap diem A: 1 2
Nhap diem B: 4 6
Nhap diem C: 7 2
A(1,2) B(4,6) C(7,2)
Loai tam giac: Tam giac can
Chu vi: 16
Dien tich: 12
Nhap vector tinh tien: 1 -1
Tam giac sau khi tinh tien: A(2,1) B(5,5) C(8,1)
Nhap ti le phong to/ thu nho: 0.5
Tam giac sau khi phong to: A(1,0.5) B(2.5,2.5) C(4,0.5)
Tam giac sau khi thu nho: A(2,1) B(5,5) C(8,1)
Nhap lan luot goc quay, tam x0, tam y0, chieu (1/2=am/duong): 270 0 0 1
A(-1,2) B(-5,5) C(-1,8)

```

Hình 9. Kiểm thử 8 (cTamGiac)

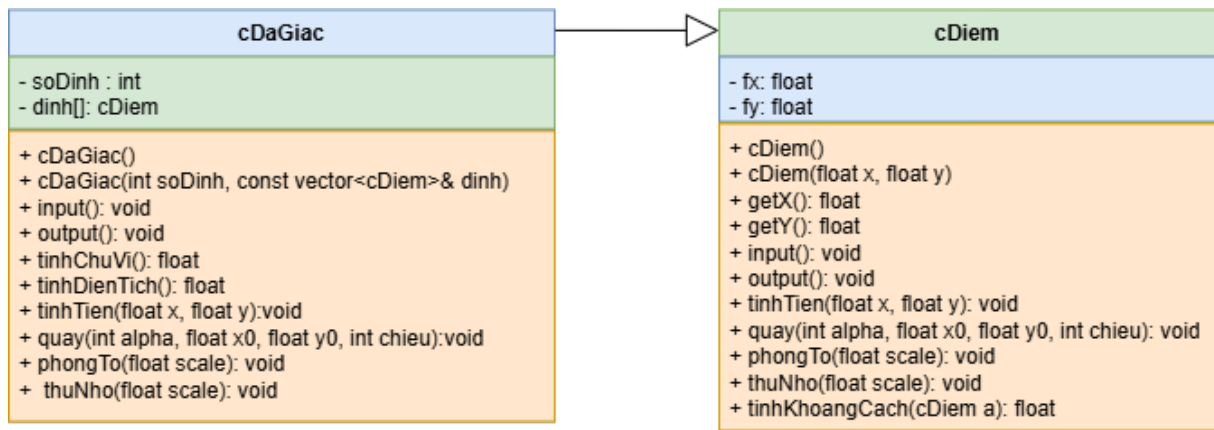
## Bài tập 2

Định nghĩa lớp cDaGiac để biểu diễn khái niệm đa giác trong mặt phẳng với các hàm thành phần tương tự như lớp TamGiac. Viết chương trình minh họa.

Nội dung diagram chương trình lớp cDaGiac:



## IT002 – Lập trình hướng đối tượng



Hình 10. Class diagram Bài 2

Nội dung code chương trình lớp đa cDaGiac:

```

#pragma once
#include <iostream>

class cDiem
{
private:
    float fx;
    float fy;
public:
    cDiem() {
        this->fx = 0;
        this->fy = 0;
    }
    cDiem(float x, float y) {
        this->fx = x;
        this->fy = y;
    }
    float getX();
    float getY();
    void input();
    void output();
    void tinhTien(float x, float y);
    void quay(int alpha, float x0, float y0, int chieu);
    void phongTo(float scale);
    void thuNho(float scale);
    float tinhKhoangCach(cDiem a);
};
    
```

Bảng 7. cDiem.h

```

#include "cDiem.h"
#include <cmath>
using namespace std;

//Định nghĩa số pi
#ifndef PI
#define PI 3.14159265358979323846
#endif

void cDiem::input() {
    cin >> fx >> fy;
}
    
```

```

void cDiem::output() {
    cout << " (" << fx << "," << fy << ") ";
}

// Tính tiền điểm
// Input: 2 số thực x, y; Output: điểm sau khi tính tiền
void cDiem::tinhTien(float x, float y) {
    fx += x;
    fy += y;
}

// Quay điểm quanh điểm (x0, y0)
// Input: 3 số nguyên alpha, x0, y0, chieu(1: chiều âm; 2: chiều dương );
// Output: điểm sau khi quay
void cDiem::quay(int alpha, float x0, float y0, int chieu) {
    float rad = float(alpha) * PI / 180;
    float dx = fx - x0;
    float dy = fy - y0;

    float fx_new, fy_new;

    if (chieu == 1) { // xoay cùng chiều kim đồng hồ(chiều âm)
        fx_new = x0 + dx * cos(rad) + dy * sin(rad);
        fy_new = y0 - dx * sin(rad) + dy * cos(rad);
    }
    else { // ngược chiều
        fx_new = x0 + dx * cos(rad) - dy * sin(rad);
        fy_new = y0 + dx * sin(rad) + dy * cos(rad);
    }

    fx = fx_new;
    fy = fy_new;
}

// Phóng to điểm
// Input: 1 số thực scale; Output: điểm sau khi phóng to
void cDiem::phongTo(float scale) {
    fx *= scale;
    fy *= scale;
}

// Thu nhỏ điểm
// Input: 1 số thực scale; Output: điểm sau khi thu nhỏ
void cDiem::thuNho(float scale) {
    fx /= scale;
    fy /= scale;
}

// Tính khoảng cách giữa 2 điểm
// Input: 1 điểm a; Output: khoảng cách giữa 2 điểm
float cDiem::tinhKhoangCach(cDiem a) {
    return sqrt(pow(fx - a.fx, 2) + pow(fy - a.fy, 2));
}

```

Bảng 8. cDiem.cpp

```

#pragma once
#include "cDiem.h"
#include <vector>
using namespace std;

class cDaGiac : public cDiem
{
private:

```

```

    int soDinh;
    vector<cDiem> dinh;
public:
    cDaGiac() : soDinh(0) {
    }

    cDaGiac(int soDinh, const vector<cDiem>& dinh) : soDinh(soDinh), dinh(dinh)
    {}

    void input();
    void output();
    float tinhChuVi();
    float tinhDienTich();
    void tinhTien(float x, float y);
    void quay(int alpha, float x0, float y0, int chieu);
    void phongTo(float scale);
    void thuNho(float scale);
};

```

Bảng 9. cDaGiac.h

```

#include "cDaGiac.h"
#include "cDiem.h"

void cDaGiac::input() {
    cout << "Nhap so luong dinh cua da giac: ";
    cin >> soDinh;

    for (int i = 0; i < soDinh; i++) {
        cout << "Nhap dinh thu " << i + 1 << " (x y):";
        cDiem tam;
        tam.input();
        dinh.push_back(cDiem(tam.getX(), tam.getY()));
    }
}

void cDaGiac::output() {
    for (int i = 0; i < soDinh; i++) {
        dinh[i].output();
    }
    cout << endl;
}

// Tinh chu vi da giac
// Input: khong co; Output: chu vi da giac
float cDaGiac::tinhChuVi() {
    float chuVi=dinh[0].tinhKhoangCach(dinh[soDinh-1]);
    for (int i = 0; i < soDinh-1; i++) {
        chuVi += dinh[i].tinhKhoangCach(dinh[i + 1]);
    }
    return chuVi;
}

// Tinh dien tich da giac
// Input: khong co; Output: dien tich da giac
float cDaGiac::tinhDienTich() {
    float dienTich = 0;
    for (int i = 0; i < soDinh; i++) {
        int j = (i + 1) % soDinh;
        dienTich += dinh[i].getX() * dinh[j].getY();
        dienTich -= dinh[j].getX() * dinh[i].getY();
    }
    return abs(dienTich)/2;
}

```

```

// Tinh tien da giac
// Input: 2 so thuc x, y; Output: da giac sau khi tinh tien
void cDaGiac::tinhTien(float x, float y) {
    for (cDiem& a : dinh)
        a.tinhTien(x, y);
    cout << "Da Giac sau khi tinh tien: ";
    output();
}

// Quay da giac
// Input: 3 so nguyen alpha, x0, y0, chieu(1: chieu am; 2: chieu duong );
// Output: da giac sau khi quay
void cDaGiac::quay(int alpha, float x0, float y0, int chieu) {
    for (cDiem& a : dinh) {
        a.quay(alpha, x0, y0, chieu);
    }
    cout << "Da giac sau khi quay: ";
    output();
}

// Phong to da giac
// Input: 1 so thuc scale; Output: da giac sau khi phong to
void cDaGiac::phongTo(float scale) {
    for (cDiem& a : dinh) {
        a.phongTo(scale);
    }
    cout << "Da giac sau khi phong to: ";
    output();
}

// Thu nho da giac
// Input: 1 so thuc scale; Output: da giac sau khi thu nho
void cDaGiac::thuNho(float scale) {
    for (cDiem& a : dinh) {
        a.thuNho(scale);
    }
    cout << "Da giac sau khi thu nho: ";
    output();
}

```

Bảng 10. cDaGiac.cpp

```

#include "cDaGiac.h"

int main()
{
    cDaGiac a;
    a.input();
    a.output();
    cout << "Chu vi: " << a.tinhChuVi() << endl;
    cout << "Dien tich: " << a.tinhDienTich() << endl;
    float x, y;
    cout << "Nhap vector tinh tien: "; cin >> x >> y;
    a.tinhTien(x, y);
    float scale;
    cout << "Nhap ti le phong to/ thu nho: "; cin >> scale;
    a.phongTo(scale);
    a.thuNho(scale);
    int alpha, chieu;
    float x0, y0;
    cout << "Nhap lan luot goc quay, tam x0, tam y0, chieu (1/2=am/duong): ";
    cin >> alpha >> x0 >> y0 >> chieu;
}

```

## IT002 – Lập trình hướng đối tượng

```
a.quay(alpha, x0, y0, chieu);
return 0;
}
```

Bảng 11. main.cpp (cDaGiac)

Nội dung kiểm thử chương trình lớp cDaGiac:

STT	Đa giác (tọa độ các đỉnh)	Vector tịnh tiến (x,y)	Tỉ lệ thu/phóng	Góc quay (°)	Tâm quay (x0,y0)	Chiều quay	Mục đích kiểm thử
1	(0,0), (4,0), (4,3), (0,3)	(1,1)	2.0	90	(0,0)	Dương	Hình chữ nhật chuẩn, test cơ bản
2	(0,0), (3,0), (4,2), (1,2)	(-1,-1)	0.5	180	(0,0)	Âm	Hình bình hành, kiểm tra thu nhỏ và quay âm
3	(0,0), (2,0), (3,2), (1,4), (- 1,2)	(0,0)	1.0	0	(0,0)	Dương	Ngũ giác đều, không biến đổi
4	(1,1), (3,1), (4,2), (3,3), (1,3), (0,2)	(2,0)	1.5	60	(2,2)	Dương	Lục giác lỗi, test phóng to và quay lệch tâm
5	(0,0), (4,0), (4,1), (1,1), (1,4), (0,4)	(0,0)	1.0	90	(2,2)	Dương	Hình chữ G, test quay quanh tâm hình học
6	(0,0), (3,0), (3,1), (1,1), (1,3), (0,3)	(-2,-2)	0.8	45	(0,0)	Âm	Hình U lỗi, test tịnh tiến, thu nhỏ, quay âm

**IT002 – Lập trình hướng đối tượng**

7	(1,1), (4,1), (5,3), (4,5), (1,5), (0,3)	(1,-1)	2.0	120	(1,1)	Dương	Lục giác đối xứng, test nhiều phép biến đổi
8	(0,0), (1,3), (3,3), (4,0), (2,- 2)	(0,2)	0.6	270	(2,2)	Âm	Ngũ giác bất kỳ, test quay và thu nhỏ phức tạp

*Bảng 12. Test cases cDaGiac*

Kiểm thử 1: Đa giác [(0,0), (4,0), (4,3), (0,3)], vector tịnh tiến(1,1), tỉ lệ thu phóng=2.0, góc quay=90, tâm quay(0,0), chiều dương

```

Nhap so luong dinh cua da giac: 4
Nhap dinh thu 1 (x y):0 0
Nhap dinh thu 2 (x y):4 0
Nhap dinh thu 3 (x y):4 3
Nhap dinh thu 4 (x y):0 3
(0,0) (4,0) (4,3) (0,3)
Chu vi: 14
Dien tích: 12
Nhap vector tinh tien: 1 1
Da Giac sau khi tinh tien: (1,1) (5,1) (5,4) (1,4)
Nhap ti le phong to/ thu nho: 2.0
Da giac sau khi phong to: (2,2) (10,2) (10,8) (2,8)
Da giac sau khi thu nho: (1,1) (5,1) (5,4) (1,4)
Nhap lan lot goc quay, tam x0, tam y0, chieu (1/2=am/duong): 90 0 0 2
Da giac sau khi quay: (-1,1) (-1,5) (-4,5) (-4,1)

```

*Hình 11. Kiểm thử 1 (cDaGiac)*

Kiểm thử 2: Đa giác [(0,0), (3,0), (4,2), (1,2)], vector tịnh tiến(-1,-1), tỉ lệ thu phóng=0.5, góc quay=180, tâm quay(0,0), chiều âm

```
Nhap so luong dinh cua da giac: 4
Nhap dinh thu 1 (x y):0 0
Nhap dinh thu 2 (x y):3 0
Nhap dinh thu 3 (x y):4 2
Nhap dinh thu 4 (x y):1 2
(0,0) (3,0) (4,2) (1,2)
Chu vi: 10.4721
Dien tích: 6
Nhap vector tinh tien: -1 -1
Da Giac sau khi tinh tien: (-1,-1) (2,-1) (3,1) (0,1)
Nhap tỉ lệ phóng to/ thu nhỏ: 0.5
Da giac sau khi phóng to: (-0.5,-0.5) (1,-0.5) (1.5,0.5) (0,0.5)
Da giac sau khi thu nhỏ: (-1,-1) (2,-1) (3,1) (0,1)
Nhap lan luot goc quay, tam x0, tam y0, chieu (1/2=am/duong): 180 0 0 1
Da giac sau khi quay: (1,1) (-2,1) (-3,-1) (-8.74228e-08,-1)
```

Hình 12. Kiểm thử 2 (cDaGiac)

Kiểm thử 3: Đa giác [(0,0), (2,0), (3,2), (1,4), (-1,2)], vector tịnh tiến(0,0), tỉ lệ thu phóng=1.0, góc quay=0, tâm quay(0,0), chiều dương

```
Nhap so luong dinh cua da giac: 5
Nhap dinh thu 1 (x y):0 0
Nhap dinh thu 2 (x y):2 0
Nhap dinh thu 3 (x y):3 2
Nhap dinh thu 4 (x y):1 4
Nhap dinh thu 5 (x y):-1 2
(0,0) (2,0) (3,2) (1,4) (-1,2)
Chu vi: 12.129
Dien tích: 10
Nhap vector tinh tien: 0 0
Da Giac sau khi tinh tien: (0,0) (2,0) (3,2) (1,4) (-1,2)
Nhap tỉ lệ phóng to/ thu nhỏ: 1.0
Da giac sau khi phóng to: (0,0) (2,0) (3,2) (1,4) (-1,2)
Da giac sau khi thu nhỏ: (0,0) (2,0) (3,2) (1,4) (-1,2)
Nhap lan luot goc quay, tam x0, tam y0, chieu (1/2=am/duong): 0 0 0 2
Da giac sau khi quay: (0,0) (2,0) (3,2) (1,4) (-1,2)
```

Hình 13. Kiểm thử 3 (cDaGiac)

Kiểm thử 4: Đa giác [(1,1), (3,1), (4,2), (3,3), (1,3), (0,2)], vector tịnh tiến(2,0), tỉ lệ thu phóng=1.5, góc quay=60, tâm quay(2,2), chiều dương

```
Nhap so luong dinh cua da giac: 6
Nhap dinh thu 1 (x y):1 1
Nhap dinh thu 2 (x y):3 1
Nhap dinh thu 3 (x y):4 2
Nhap dinh thu 4 (x y):3 3
Nhap dinh thu 5 (x y):1 3
Nhap dinh thu 6 (x y):0 2
(1,1) (3,1) (4,2) (3,3) (1,3) (0,2)
Chu vi: 9.65685
Dien tích: 6
Nhap vector tinh tien: 2 0
Da Giac sau khi tinh tien: (3,1) (5,1) (6,2) (5,3) (3,3) (2,2)
Nhap tỉ lệ phóng to/ thu nhỏ: 1.5
Da giac sau khi phóng to: (4.5,1.5) (7.5,1.5) (9,3) (7.5,4.5) (4.5,4.5) (3,3)
Da giac sau khi thu nhỏ: (3,1) (5,1) (6,2) (5,3) (3,3) (2,2)
Nhap lan luot goc quay, tam x0, tam y0, chieu (1/2=am/duong): 60 2 2 2
Da giac sau khi quay: (3.36603,2.36603) (4.36603,4.09808) (4,5.4641) (2.63397,5.09808) (1.63397,3.36603) (2,2)
```

Hình 14. Kiểm thử 4 (cDaGiac)

## IT002 – Lập trình hướng đối tượng

Kiểm thử 5: Đa giác [(0,0), (4,0), (4,1), (1,1), (1,4), (0,4)], vector tịnh tiến(0,0), tỉ lệ thu phóng=1.0, góc quay=90, tâm quay(2,2), chiều dương

```
Nhap so luong dinh cua da giac: 6
Nhap dinh thu 1 (x y):0 0
Nhap dinh thu 2 (x y):4 0
Nhap dinh thu 3 (x y):4 1
Nhap dinh thu 4 (x y):1 1
Nhap dinh thu 5 (x y):1 4
Nhap dinh thu 6 (x y):0 4
(0,0) (4,0) (4,1) (1,1) (1,4) (0,4)
Chu vi: 16
Dien tích: 7
Nhap vector tinh tien: 0 0
Da Giac sau khi tinh tien: (0,0) (4,0) (4,1) (1,1) (1,4) (0,4)
Nhap ti le phong to/ thu nho: 1.0
Da giac sau khi phong to: (0,0) (4,0) (4,1) (1,1) (1,4) (0,4)
Da giac sau khi thu nho: (0,0) (4,0) (4,1) (1,1) (1,4) (0,4)
Nhap lan luot goc quay, tam x0, tam y0, chieu (1/2=am/duong): 90 2 2 2
Da giac sau khi quay: (4,8.74228e-08) (4,4) (3,4) (3,1) (0,1) (0,-8.74228e-08)
```

Hình 15. Kiểm thử 5 (cDaGiac)

Kiểm thử 6: Đa giác [(0,0), (3,0), (3,1), (1,1), (1,3), (0,3)], vector tịnh tiến(-2,-2), tỉ lệ thu phóng=0.8, góc quay=45, tâm quay(0,0), chiều âm

```
Nhap so luong dinh cua da giac: 6
Nhap dinh thu 1 (x y):0 0
Nhap dinh thu 2 (x y):3 0
Nhap dinh thu 3 (x y):3 1
Nhap dinh thu 4 (x y):1 1
Nhap dinh thu 5 (x y):1 3
Nhap dinh thu 6 (x y):0 3
(0,0) (3,0) (3,1) (1,1) (1,3) (0,3)
Chu vi: 12
Dien tích: 5
Nhap vector tinh tien: -2 -2
Da Giac sau khi tinh tien: (-2,-2) (1,-2) (1,-1) (-1,-1) (-1,1) (-2,1)
Nhap ti le phong to/ thu nho: 0.8
Da giac sau khi phong to: (-1.6,-1.6) (0.8,-1.6) (0.8,-0.8) (-0.8,-0.8) (-0.8,0.8) (-1.6,0.8)
Da giac sau khi thu nho: (-2,-2) (1,-2) (1,-1) (-1,-1) (-1,1) (-2,1)
Nhap lan luot goc quay, tam x0, tam y0, chieu (1/2=am/duong): 45 0 0 1
Da giac sau khi quay: (-2.82843,0) (-0.707107,-2.12132) (0,-1.41421) (-1.41421,0) (0,1.41421) (-0.707107,2.12132)
```

Hình 16. Kiểm thử 6 (cDaGiac)

Kiểm thử 7: Đa giác [(1,1), (4,1), (5,3), (4,5), (1,5), (0,3)], vector tịnh tiến(1,-1), tỉ lệ thu phóng=2.0, góc quay=120, tâm quay(1,1), chiều dương



## IT002 – Lập trình hướng đối tượng

```
Nhap so luong dinh cua da giac: 6
Nhap dinh thu 1 (x y):1 1
Nhap dinh thu 2 (x y):4 1
Nhap dinh thu 3 (x y):5 3
Nhap dinh thu 4 (x y):4 5
Nhap dinh thu 5 (x y):1 5
Nhap dinh thu 6 (x y):0 3
(1,1) (4,1) (5,3) (4,5) (1,5) (0,3)
Chu vi: 14.9443
Dien tích: 16
Nhap vector tinh tien: 1 -1
Da Giac sau khi tinh tien: (2,0) (5,0) (6,2) (5,4) (2,4) (1,2)
Nhap tỉ lệ phóng to/ thu nhỏ: 2.0
Da giac sau khi phóng to: (4,0) (10,0) (12,4) (10,8) (4,8) (2,4)
Da giac sau khi thu nhỏ: (2,0) (5,0) (6,2) (5,4) (2,4) (1,2)
Nhap lan luot goc quay, tam x0, tam y0, chieu (1/2=am/duong): 120 1 1 2
Da giac sau khi quay: (1.36603,2.36603) (-0.133975,4.9641) (-2.36603,4.83013) (-3.59808,2.9641) (-2.09808,0.366025)
(0.133975,0.5)
```

Hình 17. Kiểm thử 7 (cDaGiac)

Kiểm thử 8: Đa giác [(0,0), (1,3), (3,3), (4,0), (2,-2)], vector tịnh tiến(0,2), tỉ lệ thu phóng=0.6, góc quay=270, tâm quay(2,2), chiều âm

```
Nhap so luong dinh cua da giac: 5
Nhap dinh thu 1 (x y):0 0
Nhap dinh thu 2 (x y):1 3
Nhap dinh thu 3 (x y):3 3
Nhap dinh thu 4 (x y):4 0
Nhap dinh thu 5 (x y):2 -2
(0,0) (1,3) (3,3) (4,0) (2,-2)
Chu vi: 13.9814
Dien tích: 13
Nhap vector tinh tien: 0 2
Da Giac sau khi tinh tien: (0,2) (1,5) (3,5) (4,2) (2,0)
Nhap tỉ lệ phóng to/ thu nhỏ: 0.6
Da giac sau khi phóng to: (0,1.2) (0.6,3) (1.8,3) (2.4,1.2) (1.2,0)
Da giac sau khi thu nhỏ: (0,2) (1,5) (3,5) (4,2) (2,0)
Nhap lan luot goc quay, tam x0, tam y0, chieu (1/2=am/duong): 270 2 2 1
Da giac sau khi quay: (2,0) (-1,1) (-1,3) (2,4) (4,2)
```

Hình 18. Kiểm thử 8 (cDaGiac)

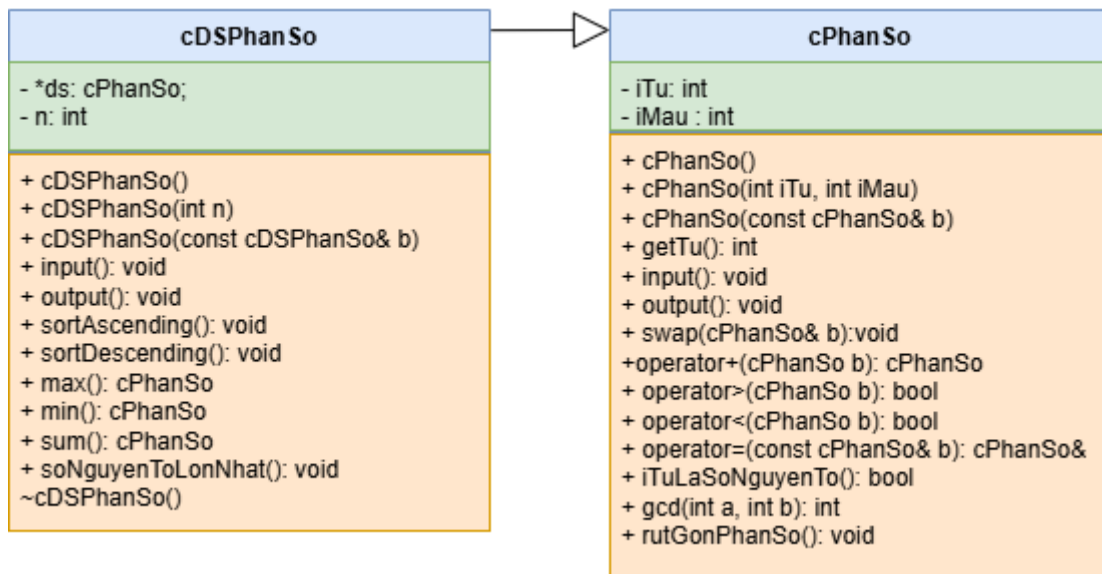
### Bài tập 3

Định nghĩa lớp cDSPhanSo (danh sách phân số) để lưu trữ và xử lý các thao tác trên mảng các phân số. Viết chương trình cho phép người dùng nhập vào danh sách các phân số

- Tính tổng các phân số
- Tìm phân số lớn nhất
- Tìm phân số nhỏ nhất
- Tìm phân số có tử là số nguyên tố lớn nhất
- Sắp xếp danh sách phân số tăng dần
- Sắp xếp danh sách phân số giảm dần

Nội dung diagram chương trình định nghĩa lớp cDSPhanSo:

## IT002 – Lập trình hướng đối tượng



Hình 19. Class diagram Bài 3

Nội dung code chương trình định nghĩa lớp cDSPhanSo:

```

#pragma once
class cPhanSo
{
private:
    int iTu;
    int iMau;
public:
    cPhanSo() {
        iTu = 0;
        iMau = 1;
    }
    cPhanSo(int iTu, int iMau) {
        this->iTu = iTu;
        this->iMau = iMau;
    }
    cPhanSo(const cPhanSo& b) {
        this->iTu = b.iTu;
        this->iMau = b.iMau;
    }
    int getTu();
    void input();
    void output();
    void swap(cPhanSo& b);
    cPhanSo operator+(cPhanSo b);
    bool operator>(cPhanSo b);
    bool operator<(cPhanSo b);
    cPhanSo& operator=(const cPhanSo& b);
    bool iTuLaSoNguyenTo();
    int gcd(int a, int b);
    void rutGonPhanSo();
};
    
```

Bảng 13. cPhanSo.h

```

#include "cPhanSo.h"
#include <iostream>
#include <cstdlib>
    
```

```

using namespace std;

int cPhanSo::getTu() {
    return iTu;
}
void cPhanSo::input() {
    do {
        cin >> iTu >> iMau;
    } while (iMau == 0);
}
void cPhanSo::output() {
    cout << iTu << "/" << iMau << " ";
}

// Hoan doi hai phan so
// Input: 2 phan so a, b; Output: hoan doi gia tri cua a va b
void cPhanSo::swap(cPhanSo& b) {
    cPhanSo temp = *this;
    *this = b;
    b = temp;
}

// Toan tu +
// Input: 2 phan so a, b; Output: phan so a + b
cPhanSo cPhanSo::operator+(cPhanSo b) {
    return cPhanSo(iTu * b.iMau + b.iTu * iMau, iMau * b.iMau);
}
// Toan tu >
// Input: 2 phan so a, b; Output: true neu a > b, false neu a <= b
bool cPhanSo::operator>(cPhanSo b) {
    return (iTu * b.iMau > b.iTu * iMau);
}

// Toan tu <
// Input: 2 phan so a, b; Output: true neu a < b, false neu a >= b
bool cPhanSo::operator<(cPhanSo b) {
    return (iTu * b.iMau < b.iTu * iMau);
}

// Toan tu =
// Input: 2 phan so a, b; Output: gan phan so b cho phan so a
cPhanSo& cPhanSo::operator=(const cPhanSo& b) {
    if (this != &b) {
        this->iTu = b.iTu;
        this->iMau = b.iMau;
    }
    return *this;
}

// Kiem tra tu so co phai la so nguyen to hay khong
// Input: phan so a; Output: true neu tu so a la so nguyen to, false neu tu so
a khong phai la so nguyen to
bool cPhanSo::iTuLaSoNguyenTo() {
    if (iTu < 2) return false;
    if (iTu == 2) return true;
    for (int i = 2; i <= sqrt(iTu); i++) {
        if (iTu % i == 0) return false;
    }
    return true;
}

int cPhanSo::gcd(int a, int b) {
    while (b != 0) {
        int temp = b;

```

```

        b = a % b;
        a = temp;
    }
    return a;
}
void cPhanSo::rutGonPhanSo() {
    int UCLN = gcd(abs(iTu), abs(iMau));
    iTu /= UCLN;
    iMau /= UCLN;
    if (iMau < 0) {
        iTu = -iTu;
        iMau = -iMau;
    }
}
}

```

Bảng 14. cPhanSo.cpp

```

#pragma once
#include "cPhanSo.h"
class cDSPhanSo :
    public cPhanSo
{
private:
    cPhanSo* ds;
    int n;
public:
    cDSPhanSo() {
        ds = nullptr;
        n = 0;
    }
    cDSPhanSo(int n) {
        this->n = n;
        ds = new cPhanSo[n];
    }
    cDSPhanSo(const cDSPhanSo& b) {
        this->n = b.n;
        this->ds = new cPhanSo[n];
        for (int i = 0; i < n; i++) {
            this->ds[i] = b.ds[i];
        }
    }
    void input();
    void output();
    void sortAscending();
    void sortDescending();
    cPhanSo max();
    cPhanSo min();
    cPhanSo sum();
    void soNguyenToLonNhat();
    ~cDSPhanSo() {
        delete[] ds;
    }
};

```

Bảng 15. cDSPhanSo.h

```

#include "cDSPhanSo.h"
#include <iostream>
#include <cmath>
using namespace std;
void cDSPhanSo::input() {
    cout << "Nhap so luong phan so: ";
}

```

```

        cin >> n;
        ds = new cPhanSo[n];
        for (int i = 0; i < n; i++) {
            cout << "Nhap phan so thu " << i + 1 << "(a b): ";
            ds[i].input();
        }
    }
    void cDSPhanSo::output() {
        for (int i = 0; i < n; i++) {
            ds[i].output();
        }
        cout << endl;
    }

    // Sap xep mang tang dan
    // Input: khong co; Output: mang sau khi sap xep tang dan
    // Thuật toán sap xep noi bot
    void cDSPhanSo::sortAscending() {
        for (int i = 0; i < n - 1; i++) {
            for (int j = i + 1; j < n; j++) {
                if (ds[i] > ds[j]) {
                    ds[i].swap(ds[j]);
                }
            }
        }
        cout << "Danh sach phan so tang dan: ";
        output();
    }

    // Sap xep mang giam dan
    // Input: khong co; Output: mang sau khi sap xep giam dan
    // Thuật toán sap xep noi bot
    void cDSPhanSo::sortDescending() {
        for (int i = 0; i < n - 1; i++) {
            for (int j = i + 1; j < n; j++) {
                if (ds[i] < ds[j]) {
                    ds[i].swap(ds[j]);
                }
            }
        }
        cout << "Danh sach phan so giam dan: ";
        output();
    }

    // Tim phan tu lon nhat trong mang
    // Input: khong co; Output: phan tu lon nhat
    cPhanSo cDSPhanSo::max() {
        cPhanSo max = ds[0];
        for (int i = 1; i < n; i++) {
            if (ds[i] > max) {
                max = ds[i];
            }
        }
        return max;
    }

    // Tim phan tu nho nhat trong mang
    // Input: khong co; Output: phan tu nho nhat
    cPhanSo cDSPhanSo::min() {
        cPhanSo min = ds[0];
        for (int i = 1; i < n; i++) {
            if (ds[i] < min) {
                min = ds[i];
            }
        }
        return min;
    }

```

```

}

// Tính tổng các phần tử trong mảng
// Input: không có; Output: tổng các phần tử trong mảng
cPhanSo cDSPhanSo::sum() {
    cPhanSo sum;
    for (int i = 0; i < n; i++) {
        sum = sum + ds[i];
    }
    sum.rutGonPhanSo();
    return sum;
}

// Tìm phần tử có iTu là số nguyên tố lớn nhất trong mảng
// Input: không có; Output: phần tử có iTu là số nguyên tố lớn nhất
void cDSPhanSo::soNguyenToLonNhat() {
    cPhanSo max;
    for (int i = 0; i < n; i++) {
        if (ds[i].iTulaSoNguyenTo()) {
            max = ds[i];
            break;
        }
    }

    for (int i = 1; i < n; i++) {
        if (ds[i].iTulaSoNguyenTo() && ds[i].getTu() > max.getTu()) {
            max = ds[i];
        }
    }
    if (max.iTulaSoNguyenTo() == true) {
        max.output();
        cout << endl;
    }
    else {
        cout << "Khong co" << endl;;
    }
}
}

```

Bảng 16. cDSPhanSo.cpp

```

#include "cDSPhanSo.h"
#include <iostream>
using namespace std;

int main()
{
    cDSPhanSo ds;
    ds.input();
    cout << "Danh sach phan so: "; ds.output();
    cout << "Phan so lon nhat: ";
    ds.max().output(); cout << endl;
    cout << "Phan so nho nhat: ";
    ds.min().output(); cout << endl;
    cout << "Tong cac phan so: ";
    ds.sum().output(); cout << endl;
    cout << "Phan so co tu la so nguyen to lon nhat: ";
    ds.soNguyenToLonNhat();
    ds.sortAscending();
    ds.sortDescending();
    return 0;
}

```

Bảng 17. main.cpp (cDSPhanSo)

## IT002 – Lập trình hướng đối tượng

Nội dung kiểm thử chương trình định nghĩa lớp cDSPhanSo:

STT	Danh sách phân số đầu vào	Kỳ vọng phân số lớn nhất	Kỳ vọng phân số nhỏ nhất	Tổng các phân số	Tử là số nguyên tố lớn nhất	Ghi chú
1	1/2, 3/4, 2/3	3/4	1/2	23/12	3/4	Các phân số dương, tử nguyên tố
2	5/2, -3/4, 7/8	5/2	-3/4	21/8	7/8	Có phân số âm
3	4/9, 10/9, 2/3	10/9	2/3	40/9	2/3	Có nhiều phân số gần nhau
4	6/5, 8/9, 14/15	6/5	8/9	412/45	Không có	Không có tử nào là số nguyên tố
5	5/7, 7/5, 3/11, 13/17	7/5	3/11	1066/385	13/17	Nhiều tử là số nguyên tố

Bảng 18. Test cases cDSPhanSO

Kiểm thử 1: Danh sách phân số [1/2, 3/4, 2/3], kỳ vọng: max=3/4, min=1/2, tổng=23/12, tử nguyên tố lớn nhất=3/4

```
Nhap so luong phan so: 3
Nhap phan so thu 1(a b): 1 2
Nhap phan so thu 2(a b): 3 4
Nhap phan so thu 3(a b): 2 3
Danh sach phan so: 1/2 3/4 2/3
Phan so lon nhat: 3/4
Phan so nho nhat: 1/2
Tong cac phan so: 23/12
Phan so co tu la so nguyen to lon nhat: 3/4
Danh sach phan so tang dan: 1/2 2/3 3/4
Danh sach phan so giam dan: 3/4 2/3 1/2
```

Hình 20. Kiểm thử 1 (cDSPhanSo)

## IT002 – Lập trình hướng đối tượng

Kiểm thử 2: Danh sách phân số  $[5/2, -3/4, 7/8]$ , kỳ vọng:  $\max=5/2$ ,  $\min=-3/4$ ,  $\text{tổng}=21/8$ , tử nguyên tố lớn nhất  $=7/8$

```
Nhap so luong phan so: 3
Nhap phan so thu 1(a b): 5 2
Nhap phan so thu 2(a b): -3 4
Nhap phan so thu 3(a b): 7 8
Danh sach phan so: 5/2 -3/4 7/8
Phan so lon nhat: 5/2
Phan so nho nhat: -3/4
Tong cac phan so: 21/8
Phan so co tu la so nguyen to lon nhat: 7/8
Danh sach phan so tang dan: -3/4 7/8 5/2
Danh sach phan so giam dan: 5/2 7/8 -3/4
```

Hình 21. Kiểm thử 2 (cDSPhanSo)

Kiểm thử 3: Danh sách phân số  $[4/9, 10/9, 2/3]$ , kỳ vọng:  $\max=10/9$ ,  $\min=4/9$ ,  $\text{tổng}=20/9$ , tử nguyên tố lớn nhất  $=2/3$

```
Nhap so luong phan so: 3
Nhap phan so thu 1(a b): 4 9
Nhap phan so thu 2(a b): 10 9
Nhap phan so thu 3(a b): 2 3
Danh sach phan so: 4/9 10/9 2/3
Phan so lon nhat: 10/9
Phan so nho nhat: 4/9
Tong cac phan so: 20/9
Phan so co tu la so nguyen to lon nhat: 2/3
Danh sach phan so tang dan: 4/9 2/3 10/9
Danh sach phan so giam dan: 10/9 2/3 4/9
```

Hình 22. Kiểm thử 3 (cDSPhanSo)

Kiểm thử 4: Danh sách phân số  $[6/5, 8/9, 14/15]$ , kỳ vọng:  $\max=6/5$ ,  $\min=8/9$ ,  $\text{tổng}=136/45$ , tử nguyên tố lớn nhất  $=(\text{không có})$



## IT002 – Lập trình hướng đối tượng

```
Nhap so luong phan so: 3
Nhap phan so thu 1(a b): 6 5
Nhap phan so thu 2(a b): 8 9
Nhap phan so thu 3(a b): 14 15
Danh sach phan so: 6/5 8/9 14/15
Phan so lon nhat: 6/5
Phan so nho nhat: 8/9
Tong cac phan so: 136/45
Phan so co tu la so nguyen to lon nhat: Khong co
Danh sach phan so tang dan: 8/9 14/15 6/5
Danh sach phan so giam dan: 6/5 14/15 8/9
```

Hình 23. Kiểm thử 4 (cDSPhanSo)

Kiểm thử 5: Danh sách phân số [5/7, 7/5, 3/11, 13/17], kỳ vọng: max=7/5, min=3/11, tổng=20628/6545, tử nguyên tố lớn nhất=13/17

```
Nhap so luong phan so: 4
Nhap phan so thu 1(a b): 5 7
Nhap phan so thu 2(a b): 7 5
Nhap phan so thu 3(a b): 3 11
Nhap phan so thu 4(a b): 13 17
Danh sach phan so: 5/7 7/5 3/11 13/17
Phan so lon nhat: 7/5
Phan so nho nhat: 3/11
Tong cac phan so: 20628/6545
Phan so co tu la so nguyen to lon nhat: 13/17
Danh sach phan so tang dan: 3/11 5/7 13/17 7/5
Danh sach phan so giam dan: 7/5 13/17 5/7 3/11
```

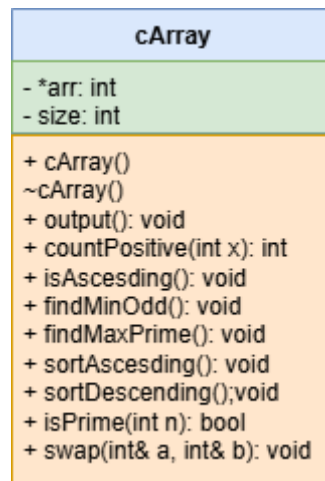
Hình 24. Kiểm thử 5 (cDSPhanSo)

### Bài tập 4

Định nghĩa lớp Mảng một chiều (cArray) để lưu trữ danh sách các số nguyên và hỗ trợ xử lý các thao tác cơ bản trên mảng một chiều:

- Tạo mảng ngẫu nhiên gồm n số nguyên
- Xuất các phần tử mảng
- Đếm số lần xuất hiện của giá trị x trong mảng
- Kiểm tra các phần tử trong mảng có tăng dần hay không
- Tìm phần tử nhỏ nhất trong mảng
- Tìm số nguyên tố lớn nhất trong mảng
- Sắp xếp các phần tử trong mảng tăng dần/giảm dần theo các thuật toán đã học trong môn Cấu trúc dữ liệu và giải thuật

Nội dung diagram của chương trình định nghĩa lớp Mảng một chiều:



Hình 25. Calss digaram Bài 4

Nội dung code của chương trình định nghĩa lớp Mảng một chiều:

```
#pragma once
#include <iostream>
#include <cstdlib> // rand(), srand()
#include <ctime> // time()

class cArray
{
private:
    int* arr;
    int size;
public:
    cArray() {
        srand(time(0));
        size = rand() % 16 + 5;
        arr = new int[size];
        for (int i = 0; i < size; i++) {
            arr[i] = rand() % 200 - 100;
        }
    }
    ~cArray() {
        delete[] arr;
    }
    void output();
    int countPositive(int x); //Dem so lan xuat hien cua x
    void isAscesding(); //Kiem tra mang co sap xep tang dan khong
    void findMinOdd(); //Tim phan tu le nho nhat
    void findMaxPrime(); //Tim phan tu nguyen to lon nhat
    void sortAscesding(); //Sap xep mang tang dan
    void sortDescending(); //Sap xep mang giam dan
    bool isPrime(int n); //Kiem tra so nguyen to
    void swap(int& a, int& b); //Hoan doi hai phan tu
};
```

Bảng 19. cArray.h

```
#include "cArray.h"
#include <iostream>
#include <cmath>
using namespace std;
void cArray::output() {
```

```

        for (int i = 0; i < size; i++) {
            cout << arr[i] << " ";
        }
        cout << endl;
    }

    // Tra ve so phan tu trong mang
    // Input: 1 so nguyen x; Output: so lan xuat hien cua x trong mang
    int cArray::countPositive(int x) {
        int count = 0;
        for (int i = 0; i < size; i++) {
            if (arr[i] == x) {
                count++;
            }
        }
        return count;
    }

    // Kiem tra mang co sap xep tang dan khong
    // Input: khong co; Output: Cac phan tu trong mang tang/giam dan
    void cArray::isAscending() {
        bool isAscending = true;
        for (int i = 0; i < size - 1; i++) {
            if (arr[i] > arr[i + 1]) {
                isAscending = false;
                break;
            }
        }
        if (isAscending) {
            cout << "Cac phan tu tang dan" << endl;
        }
        else {
            cout << "Cac phan tu khong tang dan" << endl;
        }
    }

    // Tim phan tu le nho nhat
    // Input: khong co; Output: phan tu le nho nhat
    void cArray::findMinOdd() {
        bool found = false;
        int minOdd;
        for (int i = 0; i < size; i++) {
            if (arr[i] % 2 != 0) {
                if (!found || arr[i] < minOdd) {
                    minOdd = arr[i];
                    found = true;
                }
            }
        }
        if (found) {
            cout << "Phan tu le nho nhat trong mang la: " << minOdd << endl;
        }
        else {
            cout << "Khong co phan tu le trong mang" << endl;
        }
    }

    // Tim phan tu nguyen to lon nhat
    // Input: khong co; Output: phan tu nguyen to lon nhat
    void cArray::findMaxPrime() {
        bool found = false;
        int maxPrime;
        for (int i = 0; i < size; i++) {
            if (isPrime(arr[i])) {
                if (!found || arr[i] > maxPrime) {
                    maxPrime = arr[i];
                }
            }
        }
        if (found) {
            cout << "Phan tu nguyen to lon nhat trong mang la: " << maxPrime << endl;
        }
        else {
            cout << "Khong co phan tu nguyen to trong mang" << endl;
        }
    }

```

```

        found = true;
    }
}
if (found) {
    cout << "Phan tu nguyen to lon nhat trong mang la: " << maxPrime
<< endl;
}
else {
    cout << "Khong co phan tu nguyen to trong mang" << endl;
}
}

// Sap xep mang tang dan
// Input: khong co; Output: mang sau khi sap xep tang dan
//Thuat toan sap xep noi bot
void cArray::sortAscending() {
    for (int i = 0; i < size - 1; i++) {
        for (int j = i + 1; j < size; j++) {
            if (arr[i] > arr[j]) {
                swap(arr[i], arr[j]);
            }
        }
    }
}

// Sap xep mang giam dan
// Input: khong co; Output: mang sau khi sap xep giam dan
//Thuat toan sap xep noi bot
void cArray::sortDescending() {
    for (int i = 0; i < size - 1; i++) {
        for (int j = i + 1; j < size; j++) {
            if (arr[i] < arr[j]) {
                swap(arr[i], arr[j]);
            }
        }
    }
}

// Kiem tra so nguyen to
// Input: 1 so nguyen n; Output: true neu n la so nguyen to, false neu n khong
phai la so nguyen to
bool cArray::isPrime(int n) {
    if (n <= 1) {
        return false;
    }
    if (n == 2) {
        return true;
    }
    for (int i = 2; i <= sqrt(n); i++) {
        if (n % i == 0) {
            return false;
        }
    }
    return true;
}

// Hoan doi hai phan tu
// Input: 2 so nguyen a, b; Output: hoan doi gia tri cua a va b
void cArray::swap(int& a, int& b) {
    int temp = a;
    a = b;
    b = temp;
}

```

Bảng 20. cArray.cpp

```

#include "cArray.h"
#include <iostream>

using namespace std;

int main()
{
    cArray arr;
    cout << "Số phần tử trong mảng là: " << arr.getSize() << endl;
    arr.output();
    int x;
    cout << "Nhập số x: ";
    cin >> x;
    cout << "Số lần xuất hiện của " << x << " trong mảng là: " <<
arr.countPositive(x) << endl;
    arr.isAscending();
    arr.findMinOdd();
    arr.findMaxPrime();
    arr.sortAscending();
    cout << "Mảng sau khi sắp xếp tăng dần: ";
    arr.output();
    arr.sortDescending();
    cout << "Mảng sau khi sắp xếp giảm dần: ";
    arr.output();
    return 0;
}

```

Bảng 21. main.cpp (cArray)

Nội dung kiểm thử của chương trình định nghĩa lớp Mảng một chiều:

STT	X
1	20
2	13
3	2

Bảng 22. Test cases cArray

Kiểm thử 1: Chọn x=20

```

Số phần tử trong mảng là: 9
23 -41 93 28 -91 -39 -64 54 69
Nhập số x: 20
Số lần xuất hiện của 20 trong mảng là: 0
Các phần tử không tăng dần
Phần tử nhỏ nhất trong mảng là: -91
Phần tử nguyên tố lớn nhất trong mảng là: 23
Mảng sau khi sắp xếp tăng dần: -91 -64 -41 -39 23 28 54 69 93
Mảng sau khi sắp xếp giảm dần: 93 69 54 28 23 -39 -41 -64 -91

```

Hình 26. Kiểm thử 1 (cArray)

## IT002 – Lập trình hướng đối tượng

Kiểm thử 2: Chọn x=13

```
So phan tu trong mang la: 15
-58 -88 -17 20 30 50 48 -1 57 58 19 -28 7 13 13
Nhap so x: 13
So lan xuat hien cua 13 trong mang la: 2
Cac phan tu khong tang dan
Phan tu le nho nhat trong mang la: -17
Phan tu nguyen to lon nhat trong mang la: 19
Mang sau khi sap xep tang dan: -88 -58 -28 -17 -1 7 13 13 19 20 30 48 50 57 58
Mang sau khi sap xep giam dan: 58 57 50 48 30 20 19 13 13 7 -1 -17 -28 -58 -88
```

Hình 27. Kiểm thử 2 (cArray)

Kiểm thử 3: Chọn x=2

```
So phan tu trong mang la: 8
37 -75 -100 2 -52 -49 -52 60
Nhap so x: 2
So lan xuat hien cua 2 trong mang la: 1
Cac phan tu khong tang dan
Phan tu le nho nhat trong mang la: -75
Phan tu nguyen to lon nhat trong mang la: 37
Mang sau khi sap xep tang dan: -100 -75 -52 -52 -49 2 37 60
Mang sau khi sap xep giam dan: 60 37 2 -49 -52 -52 -75 -100
```

Hình 28. Kiểm thử 3 (cArray)

## Bài tập 5

Xét đa thức theo biến x (đa thức một biến) bậc n có dạng như sau:

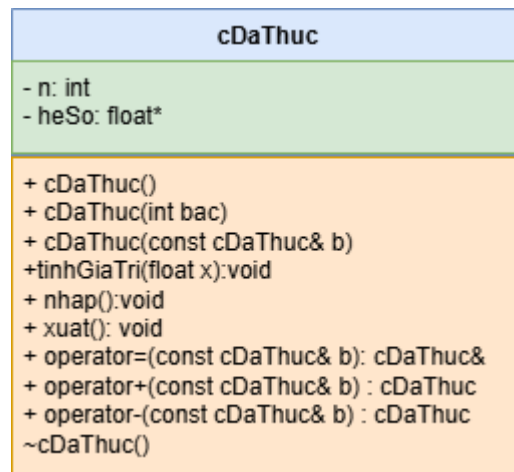
$$P(X) = ax + a_2x^{n-1} + a_3x^{n-2} + \dots + a_j$$

Trong đó: n là bậc của đa thức.  $a_1, a_2, a_3, \dots, a_j$  là các hệ số tương ứng với từng bậc của đa thức.

Định nghĩa lớp cDaThuc biểu diễn khái niệm đa thức với các thao tác sau:

- Khởi tạo một đa thức có bậc bằng 0 hoặc bậc.
- Tính giá trị của đa thức khi biết giá trị của x
- Nhập đa thức
- Xuất đa thức
- Cộng hai đa thức
- Trừ hai đa thức

Nội dung class diagram chương trình định nghĩa lớp cDaThuc:



Hình 29. Class diagram Bài 5

Nội dung class code chương trình định nghĩa lớp cDaThuc:

```
#pragma once
class cDaThuc
{
private:
    int n; // bac da thuc
    float* heSo; // mang luu cac he so
public:
    // Constructor mặc định
    cDaThuc() : n(0), heSo(new float[1] {0}) {}

    // Constructor với tham số
    cDaThuc(int bac) : n(bac), heSo(new float[bac + 1] {}) {}

    cDaThuc(const cDaThuc& b) {
        n = b.n;
        heSo = new float[n + 1];
        for (int i = 0; i <= n; i++) {
            heSo[i] = b.heSo[i];
        }
    }
    void tinhGiaTri(float x);
    void nhap();
    void xuat();
    cDaThuc& operator=(const cDaThuc& b);
    cDaThuc operator+(const cDaThuc& b) ;
    cDaThuc operator-(const cDaThuc& b) ;
    ~cDaThuc() {
        if (heSo != nullptr) {
            delete[] heSo;
            heSo = nullptr;
        }
    }
};
```

Bảng 23. cDaThuc.h

```
#include "cDaThuc.h"
#include <iostream>
using namespace std;

void cDaThuc::tinhGiaTri(float x) {
    float kq = 0;
```

```

        for (int i = 0; i <= n; i++) {
            kq += (float)heSo[i] * pow(x, i);
        }
        cout << "Gia tri da thuc tai x = " << x << " la: " << kq << endl;
    }
    void cDaThuc::nhap() {
        cout << "Nhap bac da thuc(chi so mang = he so cua x^chiso) : ";
        cin >> n;
        if (heSo != nullptr) {
            delete[] heSo; // Giải phóng bộ nhớ trước khi cấp phát mới
        }
        heSo = new float[n + 1];
        for (int i = n; i >= 0; i--) {
            cout << "Nhap he so a[" << i << "]: ";
            cin >> heSo[i];
        }
    }
    void cDaThuc::xuat() {
        bool first = true;
        for (int i = n; i >= 0; i--) {
            if (heSo[i] != 0) {
                if (!first && heSo[i] > 0) cout << " + ";
                if (heSo[i] < 0) cout << " - ";
                else if (!first && heSo[i] > 0) cout << " ";

                float absHeSo = abs(heSo[i]);
                if (i == 0 || absHeSo != 1) cout << absHeSo;
                if (i > 0) cout << "x";
                if (i > 1) cout << "^" << i;

                first = false;
            }
        }
        if (first) cout << "0"; // Nếu tất cả hệ số = 0
        cout << endl;
    }

    cDaThuc& cDaThuc::operator=(const cDaThuc& b) {
        if (this != &b) {
            delete[] heSo;
            n = b.n;
            heSo = new float[n + 1];
            for (int i = 0; i <= n; i++)
                heSo[i] = b.heSo[i];
        }
        return *this;
    }

    cDaThuc cDaThuc::operator+(const cDaThuc& b) {
        int maxBac = max(n, b.n);
        cDaThuc kq(maxBac);

        for (int i = 0; i <= maxBac; i++) {
            float heSo1 = (i <= n) ? heSo[i] : 0;
            float heSo2 = (i <= b.n) ? b.heSo[i] : 0;
            kq.heSo[i] = heSo1 + heSo2;
        }
        return kq;
    }

    cDaThuc cDaThuc::operator-(const cDaThuc& b){
        int maxBac = max(n, b.n);
        cDaThuc kq(maxBac);
        for (int i = 0; i <= maxBac; i++) {
            float heSo1 = (i <= n) ? heSo[i] : 0;

```



## IT002 – Lập trình hướng đối tượng

```

        float heSo2 = (i <= b.n) ? b.heSo[i] : 0;
        kq.heSo[i] = heSo1 - heSo2;
    }
    return kq;
}

```

Bảng 24. cDaThuc.cpp

```

#include "cDaThuc.h"
#include <iostream>
using namespace std;

int main() {
    cDaThuc a, b;
    a.nhap();
    cout << "Da thuc a: ";
    a.xuat();
    float x;
    cout << "Nhap x de tinh gia tri da thuc: ";
    cin >> x;
    a.tinhGiaTri(x);
    b.nhap();
    cout << "Da thuc b: ";
    b.xuat();
    cDaThuc c = a+b;
    cout << "Tong 2 da thuc: ";
    c.xuat();
    cDaThuc d = a-b;
    cout << "Hieu 2 da thuc: ";
    d.xuat();
    return 0;
}

```

Bảng 25. main.cpp(cDaThuc)

Nội dung class kiểm thử chương trình định nghĩa lớp cDaThuc:

Test Case	Đa thức a	x	a(x)	Đa thức b	a + b	a - b
1	$2x^2 + 3x + 1$	1	6	$x^2 + x + 1$	$3x^2 + 4x + 2$	$x^2 + 2x$
2	$5x^3 - 2x + 4$	2	40	$-5x^3 + 2x - 4$	0	$10x^3 - 4x + 8$
3	$x^4 - x + 2$	0	2	$-x^4 + x - 1$	1	$2x^4 - 2x + 3$

## IT002 – Lập trình hướng đối tượng

4	0	5	0	$x^2 + 1$	$x^2 + 1$	$-x^2 - 1$
5	$3x^2 + 0x - 3$	-1	0	$2x^2 + 1$	$5x^2 - 2$	$x^2 - 4$

Bảng 26. Test cases cDaThuc

Kiểm thử 1:  $a = 2x^2 + 3x + 1$ ,  $x = 1$ ,  $a(x) = 6$ ,  $b = x^2 + x + 1$ ,  $a + b = 3x^2 + 4x + 2$ ,  $a - b = x^2 + 2x$

```

Nhap bac da thuc(chi so mang = he so cua x^chiso) : 2
Nhap he so a[2]: 2
Nhap he so a[1]: 3
Nhap he so a[0]: 1
Da thuc a: 2x^2 + 3x + 1
Nhap x de tinh gia tri da thuc: 1
Gia tri da thuc tai x = 1 la: 6
Nhap bac da thuc(chi so mang = he so cua x^chiso) : 2
Nhap he so a[2]: 1
Nhap he so a[1]: 1
Nhap he so a[0]: 1
Da thuc b: x^2 + x + 1
Tong 2 da thuc: 3x^2 + 4x + 2
Hieu 2 da thuc: x^2 + 2x
    
```

Hình 30. Kiểm thử 1 (cDaThuc)

Kiểm thử 2:  $a = 5x^3 - 2x + 4$ ,  $x = 2$ ,  $a(x) = 40$ ,  $b = -5x^3 + 2x - 4$ ,  $a + b = 0$ ,  $a - b = 10x^3 - 4x + 8$

```

Nhap bac da thuc(chi so mang = he so cua x^chiso) : 3
Nhap he so a[3]: 5
Nhap he so a[2]: 0
Nhap he so a[1]: -2
Nhap he so a[0]: 4
Da thuc a:  $5x^3 - 2x + 4$ 
Nhap x de tinh gia tri da thuc: 2
Gia tri da thuc tai x = 2 la: 40
Nhap bac da thuc(chi so mang = he so cua x^chiso) : 3
Nhap he so a[3]: -5
Nhap he so a[2]: 0
Nhap he so a[1]: 2
Nhap he so a[0]: -4
Da thuc b:  $-5x^3 + 2x - 4$ 
Tong 2 da thuc: 0
Hieu 2 da thuc:  $10x^3 - 4x + 8$ 

```

Hình 31. Kiểm thử 2 (cDaThuc)

Kiểm thử 3:  $a = x^4 - x + 2$ ,  $x = 0$ ,  $a(x) = 2$ ,  $b = -x^4 + x - 1$ ,  $a + b = 1$ ,  $a - b = 2x^4 - 2x + 3$

```

Nhap bac da thuc(chi so mang = he so cua x^chiso) : 4
Nhap he so a[4]: 1
Nhap he so a[3]: 0
Nhap he so a[2]: 0
Nhap he so a[1]: -1
Nhap he so a[0]: 2
Da thuc a:  $x^4 - x + 2$ 
Nhap x de tinh gia tri da thuc: 0
Gia tri da thuc tai x = 0 la: 2
Nhap bac da thuc(chi so mang = he so cua x^chiso) : 4
Nhap he so a[4]: -1
Nhap he so a[3]: 0
Nhap he so a[2]: 0
Nhap he so a[1]: 1
Nhap he so a[0]: -1
Da thuc b:  $-x^4 + x - 1$ 
Tong 2 da thuc: 1
Hieu 2 da thuc:  $2x^4 - 2x + 3$ 

```

Hình 32. Kiểm thử 3 (cDaThuc)

Kiểm thử 4:  $a = 0$ ,  $x = 5$ ,  $a(x) = 0$ ,  $b = x^2 + 1$ ,  $a + b = x^2 + 1$ ,  $a - b = -x^2 - 1$

```
Nhap bac da thuc(chi so mang = he so cua x^chiso) : 0
Nhap he so a[0]: 0
Da thuc a: 0
Nhap x de tinh gia tri da thuc: 5
Gia tri da thuc tai x = 5 la: 0
Nhap bac da thuc(chi so mang = he so cua x^chiso) : 2
Nhap he so a[2]: 1
Nhap he so a[1]: 0
Nhap he so a[0]: 1
Da thuc b: x^2 + 1
Tong 2 da thuc: x^2 + 1
Hieu 2 da thuc: - x^2 - 1
```

Hình 33. Kiểm thử 4 (cDaThuc)

Kiểm thử 5:  $a = 3x^2 - 3$ ,  $x = -1$ ,  $a(x) = 0$ ,  $b = 2x^2 + 1$ ,  $a + b = 5x^2 - 2$ ,  $a - b = x^2 - 4$

```
Nhap bac da thuc(chi so mang = he so cua x^chiso) : 2
Nhap he so a[2]: 3
Nhap he so a[1]: 0
Nhap he so a[0]: -3
Da thuc a: 3x^2 - 3
Nhap x de tinh gia tri da thuc: -1
Gia tri da thuc tai x = -1 la: 0
Nhap bac da thuc(chi so mang = he so cua x^chiso) : 2
Nhap he so a[2]: 2
Nhap he so a[1]: 0
Nhap he so a[0]: 1
Da thuc b: 2x^2 + 1
Tong 2 da thuc: 5x^2 - 2
Hieu 2 da thuc: x^2 - 4
```

Hình 34. Kiểm thử 5 (cDaThuc)

## Bài tập 6

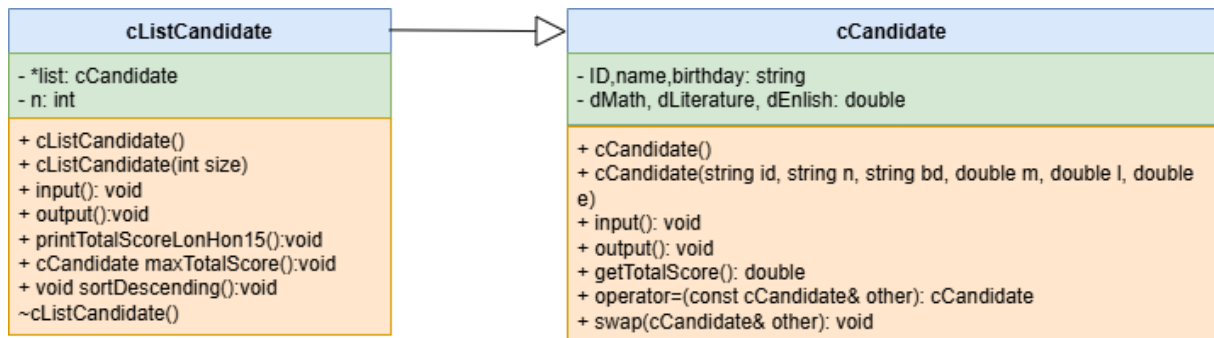
Định nghĩa lớp cCandidate (Thí sinh) gồm các thuộc tính: mã, tên, ngày tháng năm sinh, điểm thi Toán, Văn, Anh và các phương thức cần thiết.

Định nghĩa lớp cListCandidate (danh sách thí sinh) để lưu trữ và xử lý các thao tác trên danh sách thí sinh:

- Nhập vào n thí sinh (n do người dùng nhập)
- Xuất thông tin các thí sinh có tổng điểm lớn hơn 15
- Cho biết thí sinh nào có tổng điểm cao nhất
- Sắp xếp danh sách thí sinh giảm dần theo tổng điểm

Nội dung class diagram chương trình danh sách thí sinh:

## IT002 – Lập trình hướng đối tượng



Hình 35. Class diagram Bài 6

Nội dung code chương trình danh sách thí sinh:

```

#pragma once
#include <string>
#include <iostream>
using namespace std;
class cCandidate
{
private:
    string ID, name, birthDay;
    double dMath, dLiterature, dEnglish;
public:
    cCandidate() {
        ID = "";
        name = "";
        birthDay = "";
        dMath = 0;
        dLiterature = 0;
        dEnglish = 0;
    }
    cCandidate(string id, string n, string bd, double m, double l, double e)
    {
        ID = id;
        name = n;
        birthDay = bd;
        dMath = m;
        dLiterature = l;
        dEnglish = e;
    }
    void input();
    void output();
    double getTotalScore();
    cCandidate& operator=(const cCandidate& other);
    void swap(cCandidate& other);
};
    
```

Bảng 27. cCandidate.h

```

#include "cCandidate.h"
#include <iostream>
#include <string>
using namespace std;

void cCandidate::input() {
    cout << "Nhập ID: ";
    cin >> ID;
    cin.ignore();
    cout << "Nhập tên: ";
    getline(cin, name);
    cout << "Nhập ngày sinh: ";
    getline(cin, birthDay);
}
    
```

```

        do {
            cout << "Nhap diem toan: ";
            cin >> dMath;
            cout << "Nhap diem van: ";
            cin >> dLiterature;
            cout << "Nhap diem anh: ";
            cin >> dEnglish;
        } while (dMath < 0 || dMath > 10 || dLiterature < 0 || dLiterature > 10
|| dEnglish < 0 || dEnglish > 10);
    }

    void cCandidate::output() {
        cout << ID << "-" << name << "-" << birthDay << "-" << dMath << "-" <<
dLiterature << "-" << dEnglish << endl;
    }

    //Truy van tong diem thi sinh
    //Input: Khong co; Output: Tong diem
    double cCandidate::getTotalScore() {
        return dMath + dLiterature + dEnglish;
    }

    //Ham gan gia tri cho doi tuong
    //Input: Khong co; Output: Khong co
    cCandidate& cCandidate::operator=(const cCandidate& other) {
        if (this != &other) {
            ID = other.ID;
            name = other.name;
            birthDay = other.birthDay;
            dMath = other.dMath;
            dLiterature = other.dLiterature;
            dEnglish = other.dEnglish;
        }
        return *this;
    }

    //Hoan doi 2 doi tuong
    //Input: Khong co; Output: Khong co
    //Thuat toan: Tao mot doi tuong tam, gan doi tuong hien tai cho doi tuong tam,
    // gan doi tuong khac cho doi tuong hien tai, gan doi tuong tam cho doi tuong
    khac
    void cCandidate::swap(cCandidate& other) {
        cCandidate temp = *this;
        *this = other;
        other = temp;
    }
}

```

Bảng 28. cCandidate.cpp

```

#pragma once
#include "cCandidate.h"
class cListCandidate : public cCandidate
{
private:
    cCandidate* list;
    int n;
public:
    cListCandidate() {
        list = nullptr;
        n = 0;
    }
    cListCandidate(int size) {
        n = size;
        list = new cCandidate[n];
    }
}

```

```

void input();
void output();
void printTotalScoreLonHon15();
cCandidate maxTotalScore();
void sortDescending();
~cListCandidate() {
    delete[] list;
}
};

```

Bảng 29. cListCandidate.h

```

#include "cListCandidate.h"

void cListCandidate::input() {
    cout << "Nhap so luong thi sinh: ";
    cin >> n;
    if (list != nullptr) {
        delete[] list; // Giai phong bo nho truoc khi cap phat moi
    }
    list = new cCandidate[n];
    for (int i = 0; i < n; i++) {
        cout << "Nhap thi sinh thu " << i + 1 << ": " << endl;
        list[i].input();
    }
}

void cListCandidate::output() {
    for (int i = 0; i < n; i++) {
        list[i].output();
    }
}

//In ra danh sach thi sinh co tong diem lon hon 15
//Input: Khong co; Output: Danh sach thi sinh co tong diem lon hon 15
//Thuat toan: Duyet danh sach thi sinh, neu thi sinh co tong diem lon hon 15 thi in ra
void cListCandidate::printTotalScoreLonHon15() {
    cout << "Danh sach thi sinh co tong diem lon hon 15: " << endl;
    for (int i = 0; i < n; i++) {
        if (list[i].getTotalScore() > 15) {
            list[i].output();
        }
    }
}

//In ra thi sinh co tong diem cao nhat
//Input: Khong co; Output: Thi sinh co tong diem cao nhat
//Thuat toan: Duyet danh sach thi sinh, neu thi sinh co tong diem cao nhat thi in ra
cCandidate cListCandidate::maxTotalScore() {
    cCandidate maxCandidate = list[0];
    for (int i = 1; i < n; i++) {
        if (list[i].getTotalScore() > maxCandidate.getTotalScore()) {
            maxCandidate = list[i];
        }
    }
    return maxCandidate;
}

//Sap xep danh sach thi sinh theo diem giam dan
//Input: Khong co; Output: Danh sach thi sinh sau khi sap xep
//Thuat toan: Duyet danh sach thi sinh, neu thi sinh co diem cao hon thi doi cho

```

```

void cListCandidate::sortDescending() {
    for (int i = 0; i < n - 1; i++) {
        for (int j = i + 1; j < n; j++) {
            if (list[i].getTotalScore() < list[j].getTotalScore()) {
                cCandidate temp = list[i];
                list[i] = list[j];
                list[j] = temp;
            }
        }
    }
    cout << "Danh sach thi sinh sau khi sap xep: " << endl;
    output();
}

```

Bảng 30. cCandidate.cpp

```

#include "cListCandidate.h"
#include <iostream>
#include <string>
using namespace std;

void menu() {
    cListCandidate list;
    int choice;
    do {
        cout << "\n===== MENU =====\n";
        cout << "1. Nhap danh sach thi sinh\n";
        cout << "2. Xuat danh sach thi sinh\n";
        cout << "3. In thi sinh co tong diem > 15\n";
        cout << "4. In thi sinh co tong diem cao nhat\n";
        cout << "5. Sap xep danh sach theo tong diem giam dan\n";
        cout << "0. Thoat\n";
        cout << "Nhap lua chon: ";
        cin >> choice;

        switch (choice) {
            case 1:
                list.input();
                break;
            case 2:
                cout << "Danh sach thi sinh:\n";
                list.output();
                break;
            case 3:
                list.printTotalScoreLonHon15();
                break;
            case 4:
                cout << "Thi sinh co tong diem cao nhat:\n";
                list.maxTotalScore().output();
                break;
            case 5:
                list.sortDeascending();
                break;
            case 0:
                cout << "Thoat chương trình.\n";
                break;
            default:
                cout << "Lua chon khong hop le. Vui long thu lai.\n";
        }
    } while (choice != 0);
}

int main() {
    menu();
    return 0;
}

```



}

Bảng 31. *mani.cpp* (thí sinh)

Nội dung kiểm thử chương trình danh sách thí sinh:

STT	Mục tiêu kiểm thử	Dữ liệu đầu vào	Bước thực hiện	Kết quả mong đợi	Ghi chú
1	Nhập danh sách thí sinh hợp lệ	2 thí sinh với điểm từ 0–10	Chọn menu 1 → nhập thông tin	Dữ liệu được nhập và lưu trong danh sách	Đảm bảo nhập đúng định dạng
2	Xuất danh sách	Danh sách đã nhập	Chọn menu 2	Hiển thị đầy đủ thông tin các thí sinh	So sánh với dữ liệu nhập
3	In thí sinh có tổng điểm > 15	1 thí sinh tổng 16, 1 thí sinh tổng 14	Chọn menu 3	Chỉ hiển thị thí sinh có tổng > 15	Test ranh giới
4	In thí sinh có tổng điểm cao nhất	3 thí sinh, 1 thí sinh có tổng cao nhất	Chọn menu 4	Hiển thị đúng thí sinh có tổng điểm cao nhất	Kiểm tra so sánh đúng
5	Sắp xếp danh sách theo tổng điểm giảm dần	3 thí sinh có điểm khác nhau	Chọn menu 5	Danh sách hiển thị theo thứ tự giảm dần tổng điểm	So sánh điểm
6	Thoát chương trình	-	Chọn menu 0	Kết thúc chương trình	Không có lỗi
7	Nhập sai điểm (ngoài khoảng 0–10)	Nhập 11 hoặc -2	Chọn menu 1 → nhập điểm sai	Bắt nhập lại đến khi đúng	Đảm bảo vòng lặp hoạt động
8	Nhập số lượng thí sinh = 0	0	Chọn menu 1 → nhập số lượng	Bắt chọn lại, không được nhập thông tin	Xử lý hợp lý đầu vào 0

Bảng 32. *Test cases thí sinh*

```
===== MENU =====
1. Nhập danh sách thí sinh
2. Xuất danh sách thí sinh
3. In thí sinh có tổng điểm > 15
4. In thí sinh có tổng điểm cao nhất
5. Sắp xếp danh sách theo tổng điểm giảm dần
0. Thoát
Nhập lựa chọn: 1
Nhập số lượng thí sinh: 3
Nhập thí sinh thứ 1:
Nhập ID: 001
Nhập tên: Khuong Thanh Len
Nhập ngày sinh: 02/11/2006
Nhập điểm toán: 8.5
Nhập điểm văn: 9.0
Nhập điểm anh: 8.0
Nhập thí sinh thứ 2:
Nhập ID: 002
Nhập tên: Huynh Minh Tri
Nhập ngày sinh: 05/09/2006
Nhập điểm toán: 9.0
Nhập điểm văn: 8.0
Nhập điểm anh: 7.5
Nhập thí sinh thứ 3:
Nhập ID: 003
Nhập tên: Nguyen Hai Dang
Nhập ngày sinh: 15/01/2006
Nhập điểm toán: 8.5
Nhập điểm văn: 8.0
Nhập điểm anh: 8.5
```

Hình 36. Kiểm thử 1 (thí sinh)

Kiểm thử 2: Xuất danh sách (dữ liệu từ kiểm thử 1)

```
===== MENU =====
1. Nhập danh sách thí sinh
2. Xuất danh sách thí sinh
3. In thí sinh có tổng điểm > 15
4. In thí sinh có tổng điểm cao nhất
5. Sắp xếp danh sách theo tổng điểm giảm dần
0. Thoát
Nhập lựa chọn: 2
Danh sách thí sinh:
001-Khuong Thanh Len-02/11/2006-8.5-9-8
002-Huynh Minh Tri-05/09/2006-9-8-7.5
003-Nguyen Hai Dang-15/01/2006-8.5-8-8.5
```

Hình 37. Kiểm thử 2 (thí sinh)

```
Nhap lua chon: 1
Nhap so luong thi sinh: 2
Nhap thi sinh thu 1:
Nhap ID: 003
Nhap ten: Dang Hai Duong
Nhap ngay sinh: 04/08/2006
Nhap diem toan: 7
Nhap diem van: 3.5
Nhap diem anh: 3.5
Nhap thi sinh thu 2:
Nhap ID: 004
Nhap ten: Le Anh Hoang
Nhap ngay sinh: 15/03/2006
Nhap diem toan: 8
Nhap diem van: 4
Nhap diem anh: 4

===== MENU =====
1. Nhap danh sach thi sinh
2. Xuat danh sach thi sinh
3. In thi sinh co tong diem > 15
4. In thi sinh co tong diem cao nhat
5. Sap xep danh sach theo tong diem giam dan
0. Thoat
Nhap lua chon: 3
Danh sach thi sinh co tong diem lon hon 15:
004-Le Anh Hoang-15/03/2006-8-4-4
```

Hình 38. Kiểm thử 3 (thí sinh)

```
===== MENU =====
1. Nhap danh sach thi sinh
2. Xuat danh sach thi sinh
3. In thi sinh co tong diem > 15
4. In thi sinh co tong diem cao nhat
5. Sap xep danh sach theo tong diem giam dan
0. Thoat
Nhap lua chon: 4
Thi sinh co tong diem cao nhat:
001-Khuong Thanh Len-02/11/2006-8.5-9-8
```

Hình 39. Kiểm thử 4 (thí sinh)

## IT002 – Lập trình hướng đối tượng

Kiểm thử 5: Sắp xếp danh sách theo tổng điểm giảm dần (dữ liệu từ kiểm thử 1)

```
===== MENU =====
1. Nhap danh sach thi sinh
2. Xuat danh sach thi sinh
3. In thi sinh co tong diem > 15
4. In thi sinh co tong diem cao nhat
5. Sap xep danh sach theo tong diem giam dan
0. Thoat
Nhap lua chon: 5
Danh sach thi sinh sau khi sap xep:
001-Khuong Thanh Len-02/11/2006-8.5-9-8
003-Nguyen Hai Dang-15/01/2006-8.5-8-8.5
002-Huynh Minh Tri-05/09/2006-9-8-7.5
```

Hình 40. Kiểm thử 5 (thí sinh)

Kiểm thử 6: Thoát chương trình

```
===== MENU =====
1. Nhap danh sach thi sinh
2. Xuat danh sach thi sinh
3. In thi sinh co tong diem > 15
4. In thi sinh co tong diem cao nhat
5. Sap xep danh sach theo tong diem giam dan
0. Thoat
Nhap lua chon: 0
Thoat chuong trinh.

D:\VS code\OOP\THUC HANH\IT002-Lab03\IT002-Lab03\
nh.exe (process 6924) exited with code 0 (0x0).
Press any key to close this window . . .|
```

Hình 41. Kiểm thử 6 (thí sinh)

Kiểm thử 7: Nhập sai điểm (ngoài khoảng 0–10)

```
===== MENU =====
1. Nhập danh sách thi sinh
2. Xuất danh sách thi sinh
3. In thi sinh có tổng điểm > 15
4. In thi sinh có tổng điểm cao nhất
5. Sắp xếp danh sách theo tổng điểm giảm dần
0. Thoát
Nhập lựa chọn: 8
Lựa chọn không hợp lệ. Vui lòng thử lại.

===== MENU =====
1. Nhập danh sách thi sinh
2. Xuất danh sách thi sinh
3. In thi sinh có tổng điểm > 15
4. In thi sinh có tổng điểm cao nhất
5. Sắp xếp danh sách theo tổng điểm giảm dần
0. Thoát
Nhập lựa chọn: |
```

Hình 42. Kiểm thử 7 (thí sinh)

Kiểm thử 8: Nhập số thí sinh=0

```
===== MENU =====
1. Nhap danh sach thi sinh
2. Xuat danh sach thi sinh
3. In thi sinh co tong diem > 15
4. In thi sinh co tong diem cao nhat
5. Sap xep danh sach theo tong diem giam dan
0. Thoat
Nhap lua chon: 1
Nhap so luong thi sinh: 0

===== MENU =====
1. Nhap danh sach thi sinh
2. Xuat danh sach thi sinh
3. In thi sinh co tong diem > 15
4. In thi sinh co tong diem cao nhat
5. Sap xep danh sach theo tong diem giam dan
0. Thoat
Nhap lua chon:
```

Hình 43. Kiểm thử 8 (thí sinh)

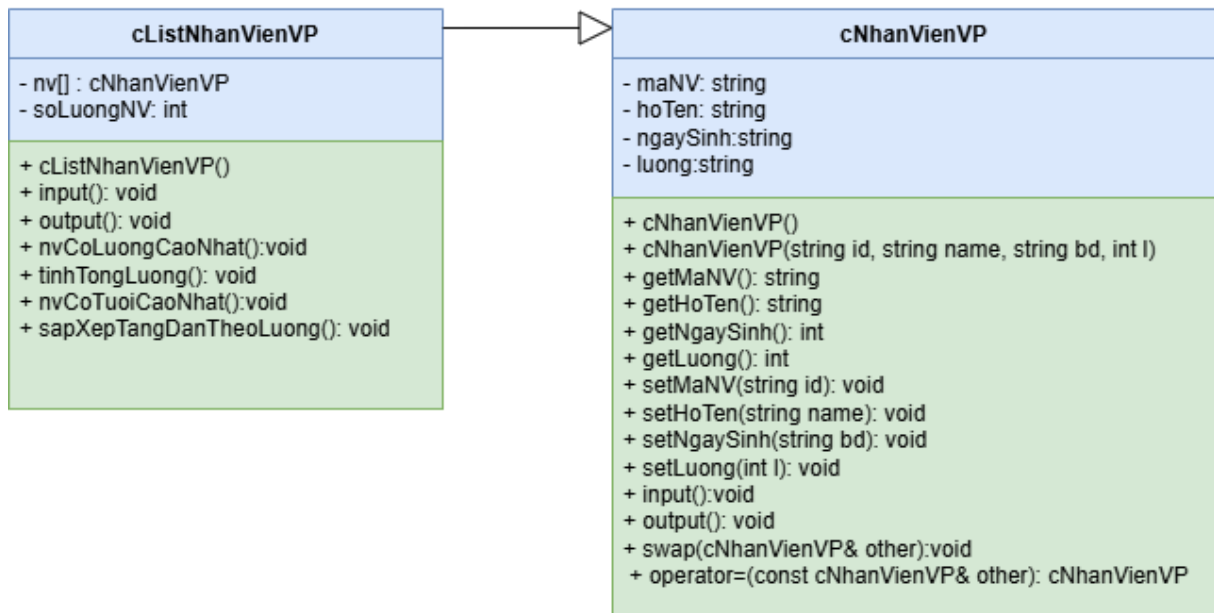
## Bài tập 7

Cài đặt lớp Nhân viên văn phòng (cNhanVienVP) cho phép lưu trữ các thông tin: mã nhân viên, họ tên, ngày sinh, lương và các thao tác cần thiết : constructor, get, set, nhập, xuất,...

Viết chương trình cho phép người dùng thực hiện các chức năng sau:

- Nhập danh sách n nhân viên văn phòng
- Xuất danh sách nhân viên văn phòng dưới dạng danh sách
- Cho biết nhân viên văn phòng nào có lương cao nhất
- Tính tổng lương mà công ty phải trả cho các nhân viên văn phòng
- Cho biết nhân viên nào có tuổi cao nhất
- Sắp xếp danh sách nhân viên văn phòng tăng dần theo lương

Nội dung class diagram chương trình cài đặt lớp Nhân viên văn phòng:



Hình 44. Class diagram Bài 7

Nội dung code chương trình cài đặt lớp Nhân viên văn phòng:

```

#pragma once
#include <string>
#include <iostream>

using namespace std;
class cNhanVienVP
{
private:
    string maNV;
    string hoTen;
    string ngaySinh;
    int luong;
public:
    cNhanVienVP() {
        this->maNV = "";
        this->hoTen = "";
        this->ngaySinh = "0/0/0";
        this->luong = 0;
    }
    cNhanVienVP(string id, string name, string bd, int l) {
        this->hoTen = name;
        this->maNV = id;
        this->ngaySinh = bd;
        this->luong = l;
    }
    string getMaNV();
    string getHoTen();
    int getNgaySinh();
    int getLuong();
    void setMaNV(string id);
    void setHoTen(string name);
    void setNgaySinh(string bd);
    void setLuong(int l);
    void input();
    void output();
    void swap(cNhanVienVP& other);
    cNhanVienVP& operator=(const cNhanVienVP& other);
}
    
```

};

Bảng 33. cNhanVienVP.h

```

#include "cNhanVienVP.h"
#include <iomanip>
#include <string>
#include <sstream>
string cNhanVienVP::getMaNV() {
    return maNV;
}
string cNhanVienVP::getHoTen() {
    return hoTen;
}
int cNhanVienVP::getNgaySinh() {
    string word, birthday = "";
    stringstream ss(ngaySinh);
    while ( getline (ss, word, '/') ) {
        birthday = word + birthday;
    }
    int kq = stoi(birthday);
    return kq;
}
int cNhanVienVP::getLuong() {
    return luong;
}
void cNhanVienVP::setMaNV(string id) {
    this->maNV = id;
}
void cNhanVienVP::setHoTen(string name) {
    this->hoTen = name;
}
void cNhanVienVP::setNgaySinh(string bd) {
    this->ngaySinh = bd;
}
void cNhanVienVP::setLuong(int l) {
    this->luong = l;
}
void cNhanVienVP::input() {
    cout << "Nhap ma nhan vien: "; cin >> maNV;
    cin.ignore();
    cout << "Nhap ho ten: "; getline(cin, hoTen);
    cout << "Nhap ngay sinh (dd/mm/yyyy): "; cin >> ngaySinh;
    do {
        cout << "Nhap luong: ";
        cin >> luong;
        if (luong < 0) {
            cout << "Loi: Luong khong duoc am. Nhap lai!\n";
        }
    } while (luong < 0);
}
void cNhanVienVP::output() {
    cout.width(10); cout << left << maNV;
    cout.width(25); cout << left << hoTen;
    cout.width(15); cout << left << ngaySinh;
    cout.width(12); cout << left << luong << endl;
}
void cNhanVienVP::swap(cNhanVienVP& other) {
    cNhanVienVP tmp;
    tmp = other;
    other = *this;
    *this = tmp;
}

```



```

cNhanVienVP& cNhanVienVP::operator=(const cNhanVienVP& other) {
    setMaNV(other.maNV);
    setHoTen(other.hoTen);
    setNgaySinh(other.ngaySinh);
    setLuong(other.luong);
    return *this;
}

```

Bảng 34. cNhanVienVP.cpp

```

#pragma once
#include "cNhanVienVP.h"
#include <vector>
class cListNhanVienVP : public cNhanVienVP
{
private:
    vector<cNhanVienVP> nv;
    int soLuongNV;
public:
    cListNhanVienVP() {
        soLuongNV = 0;
    }
    void input();
    void output();
    void nvCoLuongCaoNhat();
    void tinhTongLuong();
    void nvCoTuoiCaoNhat();
    void sapXepTangDanTheoLuong();
};

```

Bảng 35. cListNhanVienVP.h

```

#include "cListNhanVienVP.h"

void cListNhanVienVP::input() {
    cout << "Nhap so luong nhan vien: "; cin >> soLuongNV;
    for (int i = 0; i < soLuongNV; i++) {
        cout << "Nhan vien " << i + 1 << ": " << endl;
        cNhanVienVP tmp;
        tmp.input();
        nv.push_back(tmp);
    }
}

void cListNhanVienVP::output() {
    cout << "\n" << string(65, '-') << endl;
    cout.width(10); cout << left << "MaNV";
    cout.width(25); cout << left << "Ho Ten";
    cout.width(15); cout << left << "Ngay Sinh";
    cout.width(12); cout << "Luong" << endl;
    cout << string(65, '-') << endl;

    for (int i = 0; i < soLuongNV; i++) {
        nv[i].output();
    }

    cout << string(65, '-') << endl;
}

void cListNhanVienVP::nvCoLuongCaoNhat() {
    cNhanVienVP nvMax = nv[0];
    for (cNhanVienVP ds : nv) {
        if (ds.getLuong() > nvMax.getLuong()) {
            nvMax = ds;
        }
    }
}

```

```

        cout << "Nhan vien co luong cao nhat la: ";
        cout << nvMax.getMaNV() << "-" << nvMax.getHoTen() << endl;
    }
    void cListNhanVienVP::tinhTongLuong() {
        long long sum = 0;
        for (cNhanVienVP ds : nv) {
            sum += ds.getLuong();
        }
        cout << "Tong luong cong ty phai tra cho nhan vien la: " << sum << endl;
    }
    void cListNhanVienVP::nvCoTuoiCaoNhat() {
        cNhanVienVP maxTuoi = nv[0];
        for (cNhanVienVP ds : nv) {
            if (ds.getNgaySinh() < maxTuoi.getNgaySinh()) {
                maxTuoi = ds;
            }
        }
        cout << "Nhan vien co tuoi cao nhat la: ";
        cout << maxTuoi.getMaNV() << "-" << maxTuoi.getHoTen() << endl;
    }
    void cListNhanVienVP::sapXepTangDanTheoLuong() {
        for (int i = 0; i < soLuongNV - 1; i++) {
            for (int j = i + 1; j < soLuongNV; j++) {
                if (nv[j].getLuong() < nv[i].getLuong()) {
                    nv[i].swap(nv[j]);
                }
            }
        }
        cout << "Danh sach tang dan theo luong: ";
        output();
    }
}

```

Bảng 36. cListNhanVienVP.cpp

```

#include "cListNhanVienVP.h"
#include <iostream>

void menu() {
    cListNhanVienVP ds;
    int check;
    do {
        do {
            cout << "-----MENU-----" << endl;
            cout << "1.Nhap danh sach nhan vien van phong" << endl;
            cout << "2.Xuat danh sach nhan vien" << endl;
            cout << "3.Nhan vien co luong cao nhat" << endl;
            cout << "4.Tong luong cong ty phai tra cho cac nhan vien van
phong" << endl;
            cout << "5.Nhan vien co tuoi cao nhat" << endl;
            cout << "6.Sap xep danh sach nhan vien tang dan theo luong"
<< endl;
            cout << "7. Ket thuc" << endl;
            cout << "Moi ban chon: "; cin >> check;
        } while (check < 1 || check > 7);

        switch (check)
        {
            case 1:
                ds.input();
                break;
            case 2:
                ds.output();
                break;

```

## IT002 – Lập trình hướng đối tượng

```

        case 3:
            ds.nvCoLuongCaoNhat();
            break;
        case 4:
            ds.tinhTongLuong();
            break;
        case 5:
            ds.nvCoTuoiCaoNhat();
            break;
        case 6:
            ds.sapXepTangDanTheoLuong();
            break;
        default:
            exit(1);
            break;
    }
} while (check != 7);
};

int main() {
    menu();
    return 0;
}

```

Bảng 37. main.cpp (NhanVienVP)

Nội dung kiểm thử chương trình cài đặt lớp Nhân viên văn phòng:

Test cases ID	Mục tiêu	Dữ liệu vào	Kết quả mong đợi	Ghi chú
TC01	Nhập và xuất danh sách nhân viên	Nhập 2 nhân viên: NV01 - An - 01/01/1990 - 5000 NV02 - Binh - 02/02/1995 - 4500	Hiển thị đúng thông tin 2 nhân viên vừa nhập	Kiểm tra format đầu ra
TC02	Tìm nhân viên có lương cao nhất	Dùng dữ liệu ở TC01	In ra: NV01 - An là người có lương cao nhất	So sánh lương
TC03	Tính tổng lương	Dùng dữ liệu ở TC01	Tổng lương = 9500	Tính tổng đúng
TC04	Sắp xếp theo lương tăng dần	Dùng dữ liệu ở TC01	Danh sách sau sắp xếp: NV02, NV01	Kiểm tra sắp xếp
TC05	Tìm nhân viên có tuổi cao nhất	Dùng dữ liệu ở TC01	In ra: NV01 - An	So sánh tuổi
TC06	Thoát chương trình qua menu	Nhập tùy chọn 7 trong menu	Chương trình kết thúc	Không bị treo, không lỗi

Bảng 38. Test cases NhanVienVP

## IT002 – Lập trình hướng đối tượng

### Kiểm thu 1: Nhập và xuất danh sách nhân viên (TC01)

```
-----MENU-----
1.Nhap danh sach nhan vien van phong
2.Xuat danh sach nhan vien
3.Nhan vien co luong cao nhat
4.Tong luong cong ty phai tra cho cac nhan vien van phong
5.Nhan vien co tuoi cao nhat
6.Sap xep danh sach nhan vien tang dan theo luong
7. Ket thuc
Moi ban chon: 1
Nhap so luong nhan vien: 2
Nhan vien 1:
Nhap ma nhan vien: NV01
Nhap ho ten: An
Nhap ngay sinh (dd/mm/yyyy): 01/01/1990
Nhap luong: 5000
Nhan vien 2:
Nhap ma nhan vien: NV02
Nhap ho ten: Binh
Nhap ngay sinh (dd/mm/yyyy): 02/02/1995
Nhap luong: 4500
-----MENU-----
1.Nhap danh sach nhan vien van phong
2.Xuat danh sach nhan vien
3.Nhan vien co luong cao nhat
4.Tong luong cong ty phai tra cho cac nhan vien van phong
5.Nhan vien co tuoi cao nhat
6.Sap xep danh sach nhan vien tang dan theo luong
7. Ket thuc
Moi ban chon: 2
```

MaNV	Ho Ten	Ngay Sinh	Luong
NV01	An	01/01/1990	5000
NV02	Binh	02/02/1995	4500

Hình 45. Kiểm thử 1 (NhanvienVP)

### Kiểm thử 2: Tìm nhân viên có lương cao nhất (dùng dữ liệu ở TC01)

```
-----MENU-----
1.Nhap danh sach nhan vien van phong
2.Xuat danh sach nhan vien
3.Nhan vien co luong cao nhat
4.Tong luong cong ty phai tra cho cac nhan vien van phong
5.Nhan vien co tuoi cao nhat
6.Sap xep danh sach nhan vien tang dan theo luong
7. Ket thuc
Moi ban chon: 3
Nhan vien co luong cao nhat la: NV01-An
```

Hình 46. Kiểm thử 2 (NhanvienVP)

## IT002 – Lập trình hướng đối tượng

Kiểm thử 3: Tính tổng lương (dùng dữ liệu ở TC01)

```
-----MENU-----
1.Nhap danh sach nhan vien van phong
2.Xuat danh sach nhan vien
3.Nhan vien co lương cao nhất
4.Tong lương công ty phải trả cho các nhân viên văn phòng
5.Nhan vien co tuổi cao nhất
6.Sap xep danh sach nhan vien tang dan theo lương
7. Ket thuc
Moi ban chon: 4
Tong lương công ty phải trả cho nhân viên là: 9500
```

Hình 47. Kiểm thử 3 (Nhân viên VP)

Kiểm thử 4: Sắp xếp tăng dần theo lương (dùng dữ liệu ở TC01)

```
-----MENU-----
1.Nhap danh sach nhan vien van phong
2.Xuat danh sach nhan vien
3.Nhan vien co lương cao nhất
4.Tong lương công ty phải trả cho các nhân viên văn phòng
5.Nhan vien co tuổi cao nhất
6.Sap xep danh sach nhan vien tang dan theo lương
7. Ket thuc
Moi ban chon: 6
Danh sach tang dan theo lương:
-----
MaNV      Ho Ten      Ngay Sinh    Lương
-----
NV02      Binh        02/02/1995   4500
NV01      An          01/01/1990   5000
-----
```

Hình 48. Kiểm thử 4 (Nhân viên VP)

Kiểm thử 5: Tìm nhân viên có tuổi cao nhất (dùng dữ liệu ở TC01)

```
-----MENU-----
1.Nhap danh sach nhan vien van phong
2.Xuat danh sach nhan vien
3.Nhan vien co lương cao nhất
4.Tong lương công ty phải trả cho các nhân viên văn phòng
5.Nhan vien co tuổi cao nhất
6.Sap xep danh sach nhan vien tang dan theo lương
7. Ket thuc
Moi ban chon: 5
Nhan vien co tuổi cao nhất là: NV01-An
```

Hình 49. Kiểm thử 5 (Nhân viên VP)

Kiểm thử 6: Thoát chương trình

## IT002 – Lập trình hướng đối tượng

```
-----MENU-----
1.Nhap danh sach nhan vien van phong
2.Xuat danh sach nhan vien
3.Nhan vien co luong cao nhat
4.Tong luong cong ty phai tra cho cac nhan vien van phong
5.Nhan vien co tuoi cao nhat
6.Sap xep danh sach nhan vien tang dan theo luong
7. Ket thuc
Moi ban chon: 7

D:\VS code\OOP\THUC HANH\IT002-Lab03\IT002-Lab03\x64\Debug\Bai 7
Press any key to close this window . . .|
```

Hình 50. Kiểm thử 6 (NhanvienVP)

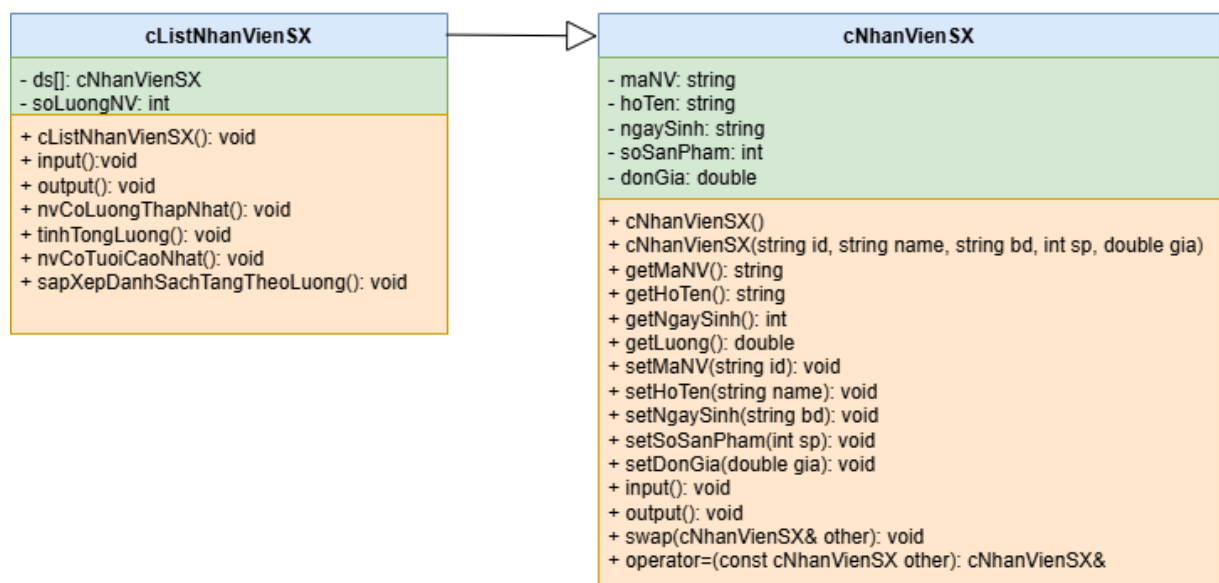
### Bài tập 8

Cài đặt lớp Nhân viên sản xuất (cNhanVienSX) cho phép lưu trữ các thông tin: mã nhân viên, họ tên, ngày sinh, số sản phẩm đã gia công, đơn giá một sản phẩm và các thao tác cần thiết: constructor, get, set, nhập, xuất, tính lương cho nhân viên = số sản phẩm đã gia công \* đơn giá một sản phẩm

Viết chương trình cho phép người dùng thực hiện các chức năng sau:

- Nhập danh sách n nhân viên sản xuất
- Xuất danh sách nhân viên sản xuất dưới dạng danh sách
- Cho biết nhân viên sản xuất nào có lương thấp nhất
- Tính tổng lương mà công ty phải trả cho các nhân viên sản xuất
- Cho biết nhân viên sản xuất nào có tuổi cao nhất
- Sắp xếp danh sách nhân viên sản xuất tăng dần theo lương

Nội dung class diagram chương trình cài đặt lớp Nhân viên sản xuất:



Hình 51. Class diagram Bài 8

## IT002 – Lập trình hướng đối tượng

Nội dung code chương trình cài đặt lớp Nhân viên sản xuất:

```
#pragma once
#include <string>
#include <iostream>

using namespace std;
class cNhanVienSX
{
private:
    string maNV;
    string hoTen;
    string ngaySinh;
    int soSanPham;
    double donGia;
public:
    cNhanVienSX() {
        this->maNV = "";
        this->hoTen = "";
        this->ngaySinh = "0/0/0";
        this->soSanPham=0;
        this->donGia = 0;
    }
    cNhanVienSX(string id, string name, string bd, int sp, double gia) {
        this->hoTen = name;
        this->maNV = id;
        this->ngaySinh = bd;
        this->soSanPham = sp;
        this->donGia = gia;
    }
    string getMaNV();
    string getHoTen();
    int getNgaySinh();
    double getLuong();
    void setMaNV(string id);
    void setHoTen(string name);
    void setNgaySinh(string bd);
    void setSoSanPham(int sp);
    void setDonGia(double gia);
    void input();
    void output();
    void swap(cNhanVienSX& other);
    cNhanVienSX& operator=(const cNhanVienSX other);
};
```

Bảng 39. cNhanVienSX.h

```
#include "cNhanVienSX.h"
#include <iomanip>
#include <string>
#include <sstream>
string cNhanVienSX::getMaNV() {
    return maNV;
}
string cNhanVienSX::getHoTen() {
    return hoTen;
}
int cNhanVienSX::getNgaySinh() {
    string word, birthday = "";
    stringstream ss(ngaySinh);
    while (getline(ss, word, '/')) {
        birthday = word + birthday;
    }
    int kq = stoi(birthday);
}
```

```

        return kq;
    }
    double cNhanVienSX::getLuong() {
        return soSanPham * donGia;
    }
    void cNhanVienSX::setMaNV(string id) {
        this->maNV = id;
    }
    void cNhanVienSX::setHoTen(string name) {
        this->hoTen = name;
    }
    void cNhanVienSX::setNgaySinh(string bd) {
        this->ngaySinh = bd;
    }
    void cNhanVienSX::setSoSanPham(int sp) {
        this->soSanPham = sp;
    }
    void cNhanVienSX::setDonGia(double gia) {
        this->donGia = gia;
    }
    void cNhanVienSX::input() {
        cout << "Nhap ma nhan vien: "; cin >> maNV;
        cin.ignore();
        cout << "Nhap ho ten: "; getline(cin, hoTen);
        cout << "Nhap ngay sinh (dd/mm/yyyy): "; cin >> ngaySinh;
        cout << "Nhap so san pham: "; cin >> soSanPham;
        cout << "Nhap don gia: "; cin >> donGia;
    }
    void cNhanVienSX::output() {
        cout.width(10); cout << left << maNV;
        cout.width(25); cout << left << hoTen;
        cout.width(15); cout << left << ngaySinh;
        cout.width(20); cout << left << soSanPham;
        cout.width(10); cout << left << donGia;
        cout.width(15); cout << left << getLuong() << endl;
    }

    void cNhanVienSX::swap(cNhanVienSX& other) {
        cNhanVienSX tmp;
        tmp = other;
        other = *this;
        *this = tmp;
    }
    cNhanVienSX& cNhanVienSX::operator=(const cNhanVienSX other) {
        setMaNV(other.maNV);
        setHoTen(other.hoTen);
        setNgaySinh(other.ngaySinh);
        setSoSanPham(other.soSanPham);
        setDonGia(other.donGia);
        return *this;
    }
}

```

Bảng 40. cNhanVienSX.cpp

```

#pragma once
#include "cNhanVienSX.h"
#include <vector>
class cListNhanVienSX : public cNhanVienSX
{
private:
    vector<cNhanVienSX> ds;
    int soLuongNV;
public:
    cListNhanVienSX() {
        soLuongNV = 0;
    }
}

```



```

    }
    void input();
    void output();
    void nvCoLuongThapNhat();
    void tinhTongLuong();
    void nvCoTuoiCaoNhat();
    void sapXepDanhSachTangTheoLuong();
};

```

Bảng 41. cListNhanVienSX.h

```

#include "cListNhanVienSX.h"
#include <iomanip>

void cListNhanVienSX::input() {
    cout << "Nhap so luong nhan vien: "; cin >> soLuongNV;
    for (int i = 0; i < soLuongNV; i++) {
        cout << "Nhan vien " << i + 1 << ": " << endl;
        cNhanVienSX tmp;
        tmp.input();
        ds.push_back(tmp);
    }
}

void cListNhanVienSX::output() {
    cout << "\n" << string(90, '-') << endl;
    cout.width(10); cout << left << "MaNV";
    cout.width(25); cout << left << "Ho Ten";
    cout.width(15); cout << left << "Ngay Sinh";
    cout.width(20); cout << left << "So Luong San Pham";
    cout.width(10); cout << left << "Don Gia";
    cout.width(15); cout << left << "Luong" << endl;
    cout << string(90, '-') << endl;

    for (int i = 0; i < soLuongNV; i++) {
        ds[i].output();
    }

    cout << string(90, '-') << endl;
}

void cListNhanVienSX::nvCoLuongThapNhat() {
    cNhanVienSX minLuong = ds[0];
    for (int i = 1; i < soLuongNV; i++) {
        if (ds[i].getLuong() < minLuong.getLuong()) {
            minLuong = ds[i];
        }
    }
    cout << "Nhan vien co luong thap nhat: ";
    cout << minLuong.getMaNV() << "-" << minLuong.getHoTen() << endl;
}

void cListNhanVienSX::tinhTongLuong() {
    int sum = 0;
    for (cNhanVienSX nv : ds) {
        sum += nv.getLuong();
    }
    cout << "Tong luong cong ty phai tra cho nhan vien: ";
    cout << sum << endl;
}

void cListNhanVienSX::nvCoTuoiCaoNhat() {
    cNhanVienSX maxTuoi = ds[0];
    for (int i = 1; i < soLuongNV; i++) {
        if (ds[i].getNgaySinh() < maxTuoi.getNgaySinh()) {
            maxTuoi = ds[i];
        }
    }
    cout << "Nhan vien co tuoi lon nhat: ";
}

```

```

        cout << maxTuoI.getMaNV() << "-" << maxTuoI.getHoTen() << endl;
    }
    void cListNhanVienSX::sapXepDanhSachTangTheoLuong() {
        for (int i = 0; i < soLuongNV - 1; i++) {
            for (int j = i+1; j < soLuongNV; j++) {
                if (ds[j].getLuong() < ds[i].getLuong()) {
                    ds[i].swap(ds[j]);
                }
            }
        }
        cout << "Danh sach sau khi sap xep: ";
        output();
    }
}

```

Bảng 42. cListNhanVienSX.cpp

```

#include "cListNhanVienSX.h"

void menu() {
    cListNhanVienSX ds;
    int check;
    do {
        do {
            cout << "-----MENU-----" << endl;
            cout << "1.Nhap danh sach nhan vien san xuat" << endl;
            cout << "2.Xuat danh sach nhan vien" << endl;
            cout << "3.Nhan vien co luong thap nhat" << endl;
            cout << "4.Tong luong cong ty phai tra cho cac nhan vien san
xuat" << endl;
            cout << "5.Nhan vien co tuoi cao nhat" << endl;
            cout << "6.Sap xep danh sach nhan vien tang dan theo luong"
<< endl;
            cout << "7. Ket thuc" << endl;
            cout << "Moi ban chon: "; cin >> check;
        } while (check < 1 || check > 7);

        switch (check)
        {
            case 1:
                ds.input();
                break;
            case 2:
                ds.output();
                break;
            case 3:
                ds.nvCoLuongThapNhat();
                break;
            case 4:
                ds.tinhTongLuong();
                break;
            case 5:
                ds.nvCoTuoicaoNhat();
                break;
            case 6:
                ds.sapXepDanhSachTangTheoLuong();
            default:
                exit(1);
                break;
        }
    } while (check != 7);
};

int main() {
    menu();
}

```

## IT002 – Lập trình hướng đối tượng

```
    return 0;
}
```

Bảng 43. main.cpp (NhanVienSX)

Nội dung kiểm thử chương trình cài đặt lớp Nhân viên sản xuất:

STT	Chức năng	Dữ liệu đầu vào (Input)	Hành động (Action)	Kết quả mong đợi (Expected Output)
1	Nhập danh sách nhân viên	2 nhân viên: NV01 - A - 01/01/1990 - 10 SP - 10000 VND NV02 - B - 01/01/1980 - 15 SP - 12000 VND	Chọn menu 1	Danh sách có 2 nhân viên được thêm vào
2	Xuất danh sách	Dữ liệu đã nhập ở Test 1	Chọn menu 2	In ra bảng 2 nhân viên với đầy đủ thông tin, có lương = $SP \times \text{Đơn giá}$
3	Tìm nhân viên có lương thấp nhất	Dữ liệu như trên	Chọn menu 3	In ra: "Nhan vien co luong thap nhat: NV01 - A" (Lương 100000 so với 180000 của NV02)
4	Tính tổng lương	Dữ liệu như trên	Chọn menu 4	In ra: "Tong luong cong ty phai tra cho nhan vien: 280000.00"
5	Tìm nhân viên có tuổi cao nhất	Dữ liệu như trên	Chọn menu 5	In ra: "Nhan vien co tuoi lon nhat: NV02 - B"
6	Sắp xếp theo lương tăng dần	NV01: 15 SP – 10000 VNĐ NV02: 12 SP – 8000 VNĐ	Chọn menu 6	In lại danh sách với thứ tự: NV02 → NV01
7	Trường hợp nhập sai menu	Nhập số 9	Nhập số không hợp lệ	In ra thông báo và yêu cầu chọn lại

Bảng 44. Test cases NhanVienSX

Kiểm thử 1: Nhập danh sách nhân viên

```

-----MENU-----
1.Nhap danh sach nhan vien san xuat
2.Xuat danh sach nhan vien
3.Nhan vien co luong thap nhat
4.Tong luong cong ty phai tra cho cac nhan vien san xuat
5.Nhan vien co tuoi cao nhat
6.Sap xep danh sach nhan vien tang dan theo luong
7. Ket thuc
Moi ban chon: 1
Nhap so luong nhan vien: 2
Nhap vien 1:
Nhap ma nhan vien: NV01
Nhap ho ten: A
Nhap ngay sinh (dd/mm/yyyy): 01/01/1990
Nhap so san pham: 10
Nhap don gia: 10000
Nhap vien 2:
Nhap ma nhan vien: NV02
Nhap ho ten: B
Nhap ngay sinh (dd/mm/yyyy): 01/01/1980
Nhap so san pham: 15
Nhap don gia: 12000

```

Hình 52. Kiểm thử 1 (NhanVienSX)

Kiểm thử 2: Xuất danh sách (Dữ liệu đã nhập ở Test 1)

```

-----MENU-----
1.Nhap danh sach nhan vien san xuat
2.Xuat danh sach nhan vien
3.Nhan vien co luong thap nhat
4.Tong luong cong ty phai tra cho cac nhan vien san xuat
5.Nhan vien co tuoi cao nhat
6.Sap xep danh sach nhan vien tang dan theo luong
7. Ket thuc
Moi ban chon: 2

```

MaNV	Ho Ten	Ngay Sinh	So Luong San Pham	Don Gia	Luong
NV01	A	01/01/1990	10	10000	100000
NV02	B	01/01/1980	15	12000	180000

Hình 53. Kiểm thử 2 (NhanVienSX)

Kiểm thử 3: Tìm nhân viên có lương thấp nhất (Dữ liệu như trên)

```
-----MENU-----
1.Nhap danh sach nhan vien san xuat
2.Xuat danh sach nhan vien
3.Nhan vien co luong thap nhat
4.Tong luong cong ty phai tra cho cac nhan vien san xuat
5.Nhan vien co tuoi cao nhat
6.Sap xep danh sach nhan vien tang dan theo luong
7. Ket thuc
Moi ban chon: 3
Nhan vien co luong thap nhat: NV01-A
MENU
```

Hình 54. Kiểm thử 3 (NhanVienSX)

Kiểm thử 4: Tính tổng lương (Dữ liệu như trên)

```
-----MENU-----
1.Nhap danh sach nhan vien san xuat
2.Xuat danh sach nhan vien
3.Nhan vien co luong thap nhat
4.Tong luong cong ty phai tra cho cac nhan vien s
5.Nhan vien co tuoi cao nhat
6.Sap xep danh sach nhan vien tang dan theo luong
7. Ket thuc
Moi ban chon: 4
Tong luong cong ty phai tra cho nhan vien: 280000
MENU
```

Hình 55. . Kiểm thử 4 (NhanVienSX)

Kiểm thử 5: Tìm nhân viên có tuổi cao nhất (Dữ liệu như trên)

```
-----MENU-----
1.Nhap danh sach nhan vien san xuat
2.Xuat danh sach nhan vien
3.Nhan vien co luong thap nhat
4.Tong luong cong ty phai tra cho cac nhan vien san xuat
5.Nhan vien co tuoi cao nhat
6.Sap xep danh sach nhan vien tang dan theo luong
7. Ket thuc
Moi ban chon: 5
Nhan vien co tuoi lon nhat: NV02-B
MENU
```

Hình 56. Kiểm thử 5 (NhanVienSX)

## IT002 – Lập trình hướng đối tượng

Kiểm thử 6: Sắp xếp theo lương tăng dần (NV01: 15 SP – 10000 VNĐ, NV02: 12 SP – 8000 VNĐ)

```
1.Nhap danh sach nhan vien san xuat
2.Xuat danh sach nhan vien
3.Nhan vien co lương thấp nhất
4.Tong lương công ty phải trả cho các nhân viên sản xuất
5.Nhan vien co tuổi cao nhất
6.Sap xep danh sach nhan vien tang dan theo lương
7. Ket thuc
Moi ban chon: 6
Danh sach sau khi sap xep:
```

MaNV	Ho Ten	Ngay Sinh	So Luong San Pham	Don Gia	Luong
NV02	B	01/01/1980	12	8000	96000
NV01	A	01/01/1990	15	10000	150000

Hình 57. Kiểm thử 6 (NhanVienSX)

Kiểm thử 7: Trường hợp nhập sai menu (Nhập số 9)

```
-----MENU-----
1.Nhap danh sach nhan vien san xuat
2.Xuat danh sach nhan vien
3.Nhan vien co lương thấp nhất
4.Tong lương công ty phải trả cho các nhân viên sản xuất
5.Nhan vien co tuổi cao nhất
6.Sap xep danh sach nhan vien tang dan theo lương
7. Ket thuc
Moi ban chon: 9
-----MENU-----
1.Nhap danh sach nhan vien san xuat
2.Xuat danh sach nhan vien
3.Nhan vien co lương thấp nhất
4.Tong lương công ty phải trả cho các nhân viên sản xuất
5.Nhan vien co tuổi cao nhất
6.Sap xep danh sach nhan vien tang dan theo lương
7. Ket thuc
Moi ban chon: |
```

Hình 58. Kiểm thử 7 (NhanVienSX)

LINK: <https://github.com/ThanhLen-IT/IT002-Lab03>