

Chương 4

SƠ LƯỢC THƯ VIỆN IOSTREAM.H

BÀI TẬP VỀ NHÀ

- Hãy khai báo và định nghĩa hàm toán tử vào và hàm toán tử ra cho các lớp đối tượng sau:
 1. Lớp phân số (CPhanSo)
 2. Lớp điểm (CDiem)
 3. Lớp ngày (CNgay)
 4. Lớp thời gian (CThoiGian)
 5. Lớp đơn thức (CDonThuc)
 6. Lớp điểm không gian (CDiemKhongGian)
 7. Lớp đường thẳng (CDuongThang)
 8. Lớp hỗn số (CHonSo)
 9. Lớp số phức (CSoPhuc)
 10. Lớp đường tròn (CDuongTron)
 11. Lớp lớp tam giác (CTamGiac)
 12. Lớp hình cầu (CHinhCau)

LỚP PHÂN SỐ

- **Bài 1 : Hãy định nghĩa toán tử vào và toán tử ra cho lớp đối tượng phân số (CPhanSo).**

LỚP PHÂN SỐ

– Khai báo lớp

```
1. class CPhanSo
2. {
3.     private:
4.         int tu;
5.         int mau;
6.     public:
7.         friend ostream& operator >>
8.             (ostream &is, CPhanSo &) ;
9.         friend ostream& operator <<
10.            (ostream &os, CPhanSo &) ;
11. } ;
```

LỚP PHÂN SỐ

– Định nghĩa toán tử vào

```
1. istream& operator >>
    (istream &is, CPhanSo &x)
2. {
3.     cout << "Nhap tu:";
4.     is >> x.tu;
5.     cout << "Nhap mau:";
6.     is >> x.mau;
7.     return is;
8. }
```

LỚP PHÂN SỐ

– Định nghĩa toán tử ra

```
1. ostream& operator <<
    (ostream &os, CPhanSo&x)
2. {
3.     os<<x.tu<<" / " <<x.mau;
4.     return os;
5. }
```

LỚP ĐIỂM

- **Bài 2 : Hãy định nghĩa toán tử vào và toán tử ra cho lớp đối tượng điểm (CDiem).**

LỚP ĐIỂM

– Khai báo lớp

```
1. class CDiem
2. {
3.     private:
4.         float x;
5.         float y;
6.     public:
7.         friend istream& operator >>
8.             (istream &is, CDiem &);
9.         friend ostream& operator <<
10.            (ostream &os, CDiem &);
11. } ;
```


LỚP ĐIỂM

– Định nghĩa toán tử vào

```
1. istream& operator >>
    (istream &is, CDiem &d)
2. {
3.     cout << "Nhap hoành do:";
4.     is >> d.x;
5.     cout << "Nhap tung do:";
6.     is >> d.y;
7.     return is;
8. }
```

LỚP ĐIỂM

– Định nghĩa toán tử ra

```
1. ostream& operator <<
    (ostream &os, CDiem&d)
2. {
3.     os<<" ("<<d.x<<" , "
        <<d.y<<" ) " ;
4.     return os;
5. }
```

LỚP NGÀY

- **Bài 3 : Hãy định nghĩa toán tử vào và toán tử ra cho lớp đối tượng ngày (CNgay).**

LỚP NGÀY

– Khai báo lớp

```
1. class CNgay
2. {
3.     private:
4.         int ng;
5.         int th;
6.         int nm;
7.     public:
8.         friend istream& operator >>
9.             (istream &is, CNgay &);
10.        friend ostream& operator <<
11.            (ostream &os, CNgay &);
12. } ;
```

LỚP NGÀY

– Định nghĩa toán tử vào

```
1. istream& operator >>
    (istream &is, CNgay &x)
2. {
3.     cout << "Nhap ngay:" ;
4.     is >> x.ngay;
5.     cout << "Nhap thang:" ;
6.     is >> x.thang;
7.     cout << "Nhap nam:" ;
8.     is >> x.nam;
9.     return is;
10. }
```

LỚP NGÀY

– Định nghĩa toán tử ra

```
1. ostream& operator <<
    (ostream &os, CNgay&x)
2. {
3.     os<<x.ng<<" / " <<x.th
        <<" / " <<x.nm;
4.     return os;
5. }
```

LỚP THỜI GIAN

– **Bài 4 : Hãy định nghĩa toán tử vào và toán tử ra cho lớp đối tượng thời gian (CThoiGian).**

LỚP THỜI GIAN

– Khai báo lớp

```
1. class CThoiGian
2. {
3.     private:
4.         int gio;
5.         int phut;
6.         int giay;
7.     public:
8.         friend ostream& operator >>
9.             (ostream &is, CThoiGian &);
10.        friend ostream& operator <<
11.            (ostream &os, CThoiGian &);
12. } ;
```


LỚP THỜI GIAN

– Định nghĩa toán tử vào

```
1. istream& operator >>
    (istream &is, CThoiGian &x)
2. {
3.     cout << "Nhap gio:";
4.     is >> x.gio;
5.     cout << "Nhap phut:";
6.     is >> x.phut;
7.     cout << "Nhap giay:";
8.     is >> x.giay;
9.     return is;
10. }
```

LỚP THỜI GIAN

– Định nghĩa toán tử ra

```
1. ostream& operator <<
    (ostream &os, CThoiGian &x)
2. {
3.     os<<x.gio<<" : "<<x.phut
        <<" : "<<x.giay;
4.     return os;
5. }
```

LỚP ĐƠN THỨC

- **Bài 5 : Hãy định nghĩa toán tử vào và toán tử ra cho lớp đối tượng đơn thức (CDonThuc).**

LỚP ĐƠN THỨC

– Khai báo lớp

```
1. class CDonThuc
2. {
3.     private:
4.         float heso;
5.         int somu;
6.     public:
7.         friend istream& operator >>
8.             (istream &is, CDonThuc &);
9.         friend ostream& operator <<
10.            (ostream &os, CDonThuc &);
11. } ;
```

LỚP ĐƠN THỨC

– Định nghĩa toán tử vào

```
1. istream& operator >>
    (istream &is, CDonThuc &x)
2. {
3.     cout << "Nhap he so:";
4.     is >> x.heso;
5.     cout << "Nhap so mu:";
6.     is >> x.somu;
7.     return is;
8. }
```

LỚP ĐƠN THỨC

– Định nghĩa toán tử ra

```
1. ostream& operator <<
    (ostream &os, CDonThuc &x)
2. {
3.     os<<x.heso<<"x^"<<x.somu;
4.     return os;
5. }
```

LỚP ĐIỂM KHÔNG GIAN

– Bài 6 : Hãy định nghĩa toán tử vào và toán tử ra cho lớp đối tượng điểm không gian (CDiemKhongGian)

LỚP ĐIỂM KHÔNG GIAN

– Khai báo lớp

```
1. class CDiemKhongGian
2. {
3.     private:
4.         float x;
5.         float y;
6.         float z;
7.     public:
8.         friend istream& operator >>
9. (istream &is,CDiemKhongGian &) ;
10.        friend ostream& operator <<
11. (ostream &os,CDiemKhongGian &) ;
12. } ;
```


LỚP ĐIỂM KHÔNG GIAN

– Định nghĩa toán tử vào

```
1. istream& operator >>
   (istream &is, CDiemKhongGian &d)
2. {
3.     cout << "Nhap x:" ;
4.     is >> d.x;
5.     cout << "Nhap y:" ;
6.     is >> d.y;
7.     cout << "Nhap z:" ;
8.     is >> d.z;
9.     return is;
10. }
```

LỚP ĐIỂM KHÔNG GIAN

– Định nghĩa toán tử ra

```
1. ostream& operator <<
   (ostream &os, CDiemKhongGian &d)
2. {
3.     os<<" ( " <<d.x<<" , "
        <<d.y<<" , " <<d.z<<" ) " ;
4.     return os;
5. }
```

LỚP ĐƯỜNG THẲNG

– Bài 7 : Hãy định nghĩa toán tử vào và toán tử ra cho lớp đối tượng đường thẳng (CDuongThang)

LỚP ĐƯỜNG THẲNG

– Khai báo lớp

```
1. class CDuongThang
2. {
3.     private:
4.         float a;
5.         float b;
6.         float c;
7.     public:
8.         friend istream& operator >>
9.             (istream &is, CDuongThang&) ;
10.        friend ostream& operator <<
11.            (ostream &os, CDuongThang&) ;
12. } ;
```

LỚP ĐƯỜNG THẲNG

– Định nghĩa toán tử vào

```
1. istream& operator >>
   (istream &is, CDuongThang&d)
2. {
3.     cout << "Nhap he so a:";
4.     is >> d.a;
5.     cout << "Nhap he so b:";
6.     is >> d.b;
7.     cout << "Nhap he so c:";
8.     is >> d.c;
9.     return is;
10. }
```

LỚP ĐƯỜNG THẲNG

– Định nghĩa toán tử ra

```
1. ostream& operator <<
   (ostream &os, CDuongThang&d)
2. {
3.     os<<d.a<<"x + "
       <<d.b<<"y + "
       <<d.c<<" = 0" ;
4.     return os;
5. }
```

LỚP HỖN SỐ

– Bài 8 : Hãy định nghĩa toán tử vào và toán tử ra cho lớp đối tượng hỗn số (CHonSo)

LỚP HỒN SỐ

– Khai báo lớp

```
1. class CHonSo
2. {
3.     private:
4.         int nguyen;
5.         int tu;
6.         int mau;
7.     public:
8.         friend istream& operator >>
9.             (istream &is, CHonSo&);
10.        friend ostream& operator <<
11.            (ostream &os, CHonSo&);
12. } ;
```


LỚP HỒN SỐ

– Định nghĩa toán tử vào

```
1. istream& operator >>
   (istream &is, CHonSo&x)
2. {
3.     cout << "Nhap phan nguyen: " ;
4.     is >> x.nguyen;
5.     cout << "Nhap tu: " ;
6.     is >> x.tu;
7.     cout << "Nhap mau: " ;
8.     is >> x.mau;
9.     return is;
10. }
```

LỚP HỒN SỐ

– Định nghĩa toán tử ra

```
1. ostream& operator <<
   (ostream &os, CHonSo&x)
2. {
3.     os<<"Phan nguyen:"<<x.nguyen;
4.     os<<"Tu:" << x.tu;
5.     os<<"Mau:" << x.mau;
6.     return os;
7. }
```

LỚP SỐ PHỨC

– Bài 9 : Hãy định nghĩa toán tử vào và toán tử ra cho lớp đối tượng số phức (CSoPhuc)

LỚP SỐ PHỨC

– Khai báo lớp

```
1. class CSoPhuc
2. {
3.     private:
4.         float thuc;
5.         float ao;
6.     public:
7.         friend ostream& operator >>
8.             (ostream &is, CSoPhuc&) ;
9.         friend ostream& operator <<
10.            (ostream &os, CSoPhuc&) ;
11. } ;
```

LỚP SỐ PHỨC

– Định nghĩa toán tử vào

```
1. istream& operator >>
   (istream &is, CHonSo&x)
2. {
3.     cout << "Nhap thuc:" ;
4.     is >> x.thuc;
5.     cout << "Nhap ao:" ;
6.     is >> x.ao;
7.     return is;
8. }
```

LỚP SỐ PHỨC

– Định nghĩa toán tử ra

```
1. ostream& operator <<
   (ostream &os, CHonSo&x)
2. {
3.     os<<"\nThuc:"<<x.thuc;
4.     os<<"\nAo:"  << x.ao;
5.     return os;
6. }
```

LỚP ĐƯỜNG TRÒN

– Bài 10 : Hãy định nghĩa toán tử vào và toán tử ra cho lớp đối tượng đường tròn (CDuongTron)

LỚP ĐƯỜNG TRÒN

– Khai báo lớp CDiem

```
1. class CDiem
2. {
3.     private:
4.         float x;
5.         float y;
6.     public:
7.         friend istream& operator >>
8.             (istream &is, CDiem &);
9.         friend ostream& operator <<
10.            (ostream &os, CDiem &);
11. } ;
```


LỚP ĐƯỜNG TRÒN

– Khai báo lớp

```
1. class CDuongTron
2. {
3.     private:
4.         CDiem tam;
5.         float bankinh;
6.     public:
7.         friend istream& operator >>
8.             (istream &is, CDuongTron&) ;
9.         friend ostream& operator <<
10.            (ostream &os, CDuongTron&) ;
11. } ;
```

LỚP ĐƯỜNG TRÒN

– Định nghĩa toán tử vào

```
1. istream& operator >>
    (istream &is, CDiem &d)
2. {
3.     cout << "Nhap x:" ;
4.     is >> d.x;
5.     cout << "Nhap y:" ;
6.     is >> d.y;
7.     return is;
8. }
```

LỚP ĐƯỜNG TRÒN

– Định nghĩa toán tử ra

```
1. ostream& operator <<
    (ostream &os, CDiem&d)
2. {
3.     os<<" ("<<d.x<<" , "
        <<d.y<<" ) " ;
4.     return os;
5. }
```

LỚP ĐƯỜNG TRÒN

– Định nghĩa toán tử vào

```
1. istream& operator >>  
   (istream &is, CDuongTron&x)  
2. {  
3.     cout << "\nTam : " ;  
4.     is >> x.tam;  
5.     cout << " \nBan kinh:" ;  
6.     is >> x.bankinh;  
7.     return is;  
8. }
```

LỚP ĐƯỜNG TRÒN

– Định nghĩa toán tử ra

```
1. ostream& operator <<
   (ostream &os, CDuongTron&x)
2. {
3.     os<<"\nTam:" << x.tam;
4.     os<<"\nR:" << x.bankinh;
5.     return os;
6. }
```

LỚP TAM GIÁC

– Bài 11 : Hãy định nghĩa toán tử vào và toán tử ra cho lớp đối tượng tam giác (CTamGiac)

LỚP TAM GIÁC

– Khai báo lớp CDiem

```
1. class CDiem
2. {
3.     private:
4.         float x;
5.         float y;
6.     public:
7.         friend istream& operator >>
8.             (istream &is, CDiem &);
9.         friend ostream& operator <<
10.            (ostream &os, CDiem &);
11. } ;
```

LỚP TAM GIÁC

– Khai báo lớp

```
1. class CTamGiac
2. {
3.     private:
4.         CDiem tam;
5.         float bankinh;
6.     public:
7.         friend ostream& operator >>
8.             (ostream &is,CTamGiac&);
9.         friend ostream& operator <<
10.            (ostream &os,CTamGiac&);
11. } ;
```


LỚP TAM GIÁC

– Định nghĩa toán tử vào

```
1. istream& operator >>
    (istream &is, CDiem &d)
2. {
3.     cout << "Nhap x:" ;
4.     is >> d.x;
5.     cout << "Nhap y:" ;
6.     is >> d.y;
7.     return is;
8. }
```

LỚP TAM GIÁC

– Định nghĩa toán tử ra

```
1. ostream& operator <<
    (ostream & os, CDiem& d)
2. {
3.     os<<" (" <<d.x<<" , "
        <<d.y<<" ) " ;
4.     return os;
5. }
```

LỚP TAM GIÁC

– Định nghĩa toán tử vào

```
1. istream& operator >>
   (istream & is, CTamGiac& x)
2. {
3.     cout << "\nDiem A :";
4.     is >> x.A;
5.     cout << "\nDiem B:";
6.     is >> x.B;
7.     cout << "\nDiem C:";
8.     is >> x.C;
9.     return is;
10. }
```

LỚP TAM GIÁC

– Định nghĩa toán tử ra

```
1. ostream& operator <<
   (ostream & os, CTamGiac& x)
2. {
3.     os<<"\nA:" << x.A;
4.     os<<"\nB:" << x.B;
5.     os<<"\nC:" << x.C;
6.     return os;
7. }
```

LỚP HÌNH CẦU

- Bài 12 : Hãy định nghĩa toán tử vào và toán tử ra cho lớp đối tượng hình cầu (CHìnhCau)

LỚP HÌNH CẦU

– Khai báo lớp

```
1. class CDiemKhongGian
2. {
3.     private:
4.         float x;
5.         float y;
6.         float z;
7.     public:
8.         friend istream& operator >>
9.         (istream &is, CDiemKhongGian &) ;
10.        friend ostream& operator <<
11.        (ostream &os, CDiemKhongGian &) ;
12. } ;
```

LỚP HÌNH CẦU

– Khai báo lớp

```
1. class CHinhCau
2. {
3.     private:
4.         CDiemKhongGian tam;
5.         float bankinh;
6.     public:
7.         friend istream& operator >>
8.             (istream &is, CHinhCau&);
9.         friend ostream& operator <<
10.            (ostream &os, CHinhCau&);
11. } ;
```

LỚP HÌNH CẦU

– Định nghĩa toán tử vào

```
1. istream& operator >>
   (istream &is, CDiemKhongGian &d)
2. {
3.     cout << "Nhap x:" ;
4.     is >> d.x;
5.     cout << "Nhap y:" ;
6.     is >> d.y;
7.     cout << "Nhap z:" ;
8.     is >> d.z;
9.     return is;
10. }
```


LỚP HÌNH CẦU

– Định nghĩa toán tử ra

```
1. ostream& operator <<
   (ostream &os, CDiemKhongGian &d)
2. {
3.     os<<" ( " <<d.x<<" , "
        <<d.y<<" , " <<d.z<<" ) " ;
4.     return os;
5. }
```

LỚP HÌNH CẦU

– Định nghĩa toán tử vào

```
1. istream& operator >>
   (istream & is, CHinhCau& x)
2. {
3.     cout << "\nNhap tam :";
4.     is >> x.tam;
5.     cout << "\nBan kinh:";
6.     is >> x.bankinh;
7.     return is;
8. }
```

LỚP HÌNH CẦU

– Định nghĩa toán tử ra

```
1. ostream& operator <<
   (ostream & os, CHinhCau& x)
2. {
3.     os<<"\nTam:" << x.tam;
4.     os<<"\nBan kinh:"
       << x.bankinh;
5.     return os;
6. }
```