第一章

5：

```
#include <iostream>
using namespace std;
int main()
{
  return 0;
}
```

6：

```
#include <iostream>
using namespace std;
int main()
{
  int a,b,c;
  a=10;
  b=23;
  c=a+b;

  cout<<c;
  cout<<endl;
  return 0;
}
```

7：

```
#include <iostream>
using namespace std;
int main()
{
  int a,b,c;
  int f(int x,int y,int z);
  cin>>a>>b>>c;
  c=f(a,b,c);
  cout<<c<<endl;
  return 0;
}

int f(int x,int y,int z)
```

```
{
   int m;
if (x<y) m=x;
     else m=y;
   if (z<m) m=z;
   return(m);
}
```

8:

```
#include <iostream>
using namespace std;
int main()
{
  int a,b,c;
  cin>>a>>b;
  c=a+b;

  return 0;
}
```

9:

```
#include <iostream>
using namespace std;
int main()
{int add(int x,int y);
  int a,b,c;
  cin>>a>>b;
  c=add(a,b);

  return 0;
}

int add(int x,int y)
{int c;
  c=x+y;
  return(c);
}
```

10:

```
#include <iostream>
using namespace std;
int main()
{void sort(int x,int y,int z);
```

```
int x,y,z;
  cin>>x>>y>>z;
  sort(x,y,z);
  return 0;
}

void sort(int x, int y, int z)
{
  int temp;
  if (x>y) {temp=x;x=y;y=temp;}        //{内 3 个语句的作用是将 x 和 y 的值互换)
  if (z<x)   cout<<z<<','<<x<<','<<y<<endl;
    else if (z<y) cout<<x<<','<<z<<','<<y<<endl;
            else cout<<x<<','<<y<<','<<z<<endl;
}
```

11:

```
#include <iostream>
using namespace std;
int main()
{int max(int a,int b,int c=0);
  int a,b,c;
  cin>>a>>b>>c;


  return 0;
}

int max(int a,int b,int c)
{if(b>a) a=b;
  if(c>a) a=c;
  return a;
}
```

12:

```
#include    <iostream>
using namespace std;
int main()
{
    void change(int ,int    ) ;
    int a,b;
    cin>>a>>b;
    if(a<b) change(a,b);

    return 0;
}
```

```
void change(int ,int )
{
   int r1,r2,temp;

   temp=r1;
   r1=r2;
   r2=temp;

}
```

13：

```
#include <iostream>
using namespace std;
int main()
{void sort(int &,int &,int &);
  int a,b,c,a1,b1,c1;

  cin>>a>>b>>c;
  a1=a;b1=b;c1=c;
  sort(a1,b1,c1);


  return 0;
}
void sort(int &i,int &j,int &k)
{ void change(int &,int &);
if (i>j) change(i,j);
   if (i>k) change(i,k);
   if (j>k) change(j,k);
}
void change(int &x,int &y)
{ int temp;
   temp=x;
   x=y;
   y=temp;
}
```

14：

```
#include <iostream>
#include <string>
using namespace std;
int main()
```

```
    s1=s1+s2;

    return 0;
}
```

15：

```
#include <iostream>
#include <string>
using namespace std;
int main()
{ string str;
    int i,n;
    char temp;
    cin>>str;
    n=str.size();
    for(i=0;i<n/2;i++)
    {temp=str[i];str[i]=str[n-i-1];str[n-i-1]=temp;}
    cout<<str<<endl;
return 0;
}
```

16：

```
#include <iostream>
#include <string>
using namespace std;
int main()
{ int i;
    void sort(string []);
    sort(str);

    for(i=0;i<5;i++)

    cout<<endl;
    return 0;
}

void sort(string s[])
{int i,j;
  string t;
  for (j=0;j<5;j++)
```

```
        for(i=0;i<5-j;i++)
             if (s[i]>s[i+1])
               {t=s[i];s[i]=s[i+1];s[i+1]=t;}
  }17: #include <iostream>
#include <string>
using namespace std;
int main()
{
   long    c[5]={10100,-123567, 1198783,-165654, 3456};
   int a[5]={1,9,0,23,-45};
   float b[5]={2.4, 7.6, 5.5, 6.6, -2.3 };
   void sort(int []);
void sort(float []);
   void sort(long []);
   sort(a);
   sort(b);
   sort(c);
   return 0;
}

void sort(int a[])
{int i,j,t;
  for (j=0;j<5;j++)
        for(i=0;i<5-j;i++)
             if (a[i]>a[i+1])
               {t=a[i];a[i]=a[i+1];a[i+1]=t;}

  for(i=0;i<5;i++)

  cout<<endl<<endl;
  }

void sort(long a[])
{int i,j;
  long t;
  for (j=0;j<5;j++)
        for(i=0;i<5-j;i++)
             if (a[i]>a[i+1])
               {t=a[i];a[i]=a[i+1];a[i+1]=t;}

  for(i=0;i<5;i++)

    cout<<endl<<endl;
}
```

```
void sort(float a[])
{int i,j;
  float t;
  for (j=0;j<5;j++)
        for(i=0;i<5-j;i++)
             if (a[i]>a[i+1])
               {t=a[i];a[i]=a[i+1];a[i+1]=t;}

  for(i=0;i<5;i++)

    cout<<endl<<endl;
}
```

```
18: #include <iostream>
#include <string>
using namespace std;
template <typename T>
void sort(T a[])
{int i,j,min;
  T t;
  for(i=0;i<5;i++)
    {min=i;
         for (j=i+1;j<5;j++)
         if(a[min]>a[j]) min=j;
      t=a[i]; a[i]=a[min]; a[min]=t;
    }

  for(i=0;i<5;i++)

  cout<<endl<<endl;
}

int main()
{ int a[5]={1,9,0,23,-45};
    float b[5]={2.4, 7.6, 5.5, 6.6, -2.3 };
    long c[5]={10100,-123567, 1198783,-165654, 3456};
    sort(a);
    sort(b);
    sort(c);
    return 0;
}
```

第二章

1

```
#include <iostream>
using namespace std;
class Time
  {
public:
    void set_time();
    void show_time();
private:                          //成员改为公用的
        int hour;
        int minute;
        int sec;

   };

void Time::set_time()                //在 main 函数之前定义
  {
    cin>>hour;
    cin>>minute;
    cin>>sec;
  }

void Time::show_time()                //在 main 函数之前定义
  {

  }

int main()
{Time t1;
t1.set_time();
  t1.show_time();
  return 0;
}
```

2：

```
#include <iostream>
using namespace std;
class Time
    {public:
        void set_time(void)
```

```
        {cin>>hour;
          cin>>minute;
          cin>>sec;
        }
      void show_time(void)


    private: int hour;
        int minute;
        int sec;
};

Time t;
int main()
  {
    t.set_time();
    t.show_time();
    return 0;
  }
  3:
#include <iostream>
using namespace std;
class Time
    {public:
        void set_time(void);
        void show_time(void);
      private:
        int hour;
        int minute;
        int sec;
    };

    void Time::set_time(void)
        {cin>>hour;
          cin>>minute;
          cin>>sec;
        }

void Time::show_time(void)


    Time t;
    int main()
    { t.set_time();
```

```
    t.show_time();
    return 0;
}
```

4：

```
//xt2-4-1.cpp(main.cpp)
#include <iostream>
using namespace std;

int main()
{Student stud;
  stud.set_value();
  stud.display();
  return 0;
}
```

```
//xt2-4-2.cpp( 即 student.cpp)
                                                    //在此文件中进行函数的定义
#include <iostream>
using namespace std;                                //不要漏写此行
void Student::display( )




}

void Student::set_value()
{ cin>>num;
cin>>name;
    cin>>sex;
}
```

5：

```
//xt2-5-1.cpp(file1.cpp)
#include <iostream>

int main()
  {Array_max   arrmax;
    arrmax.set_value();
    arrmax.max_value();
    arrmax.show_value();
    return 0;
  }

//xt2-5-2.cpp(arraymax.cpp)
#include <iostream>
```

```
using namespace std;

void Array_max::set_value()
  { int i;
     for (i=0;i<10;i++)
         cin>>array[i];
  }

void Array_max::max_value()
  {int i;
    max=array[0];
    for (i=1;i<10;i++)
      if(array[i]>max) max=array[i];
    }

void Array_max::show_value()

  }
```

6：解法一

```
#include <iostream>
using namespace std;
class Box
{public:
    void get_value();
    float volume();
    void display();
  public:
    float lengh;
    float width;
    float height;
    } ;

void Box::get_value()

    cin>>lengh;
    cin>>width;
    cin>>height;
}

float Box::volume()
{ return(lengh*width*height);}
```

```
void Box::display()
{ cout<<volume()<<endl;}

int main()
{Box box1,box2,box3;
  box1.get_value();

  box1.display();
  box2.get_value();

  box2.display();
  box3.get_value();

  box3.display();
  return 0;
}
```

解法二：

```
#include <iostream>
using namespace std;
class Box
{public:
    void get_value();
    void volume();
    void display();
  public:
    float lengh;
    float width;
    float height;
    float vol;
    };

   void Box::get_value()

       cin>>lengh;
       cin>>width;
       cin>>height;
   }

   void Box::volume()
{ vol=lengh*width*height;}

   void Box::display()
   { cout<<vol<<endl;}
```

```
int main()
{Box box1,box2,box3;
  box1.get_value();
  box1.volume();

  box1.display();
  box2.get_value();
  box2.volume();

  box2.display();
  box3.get_value();
  box3.volume();

  box3.display();
  return 0;
}
```

第三章

2:

```
#include <iostream>
using namespace std;
class Date
  {public:
      Date(int,int,int);
      Date(int,int);
      Date(int);
      Date();
      void display();
    private:
      int month;
      int day;
      int year;
    };

  Date::Date(int m,int d,int y):month(m),day(d),year(y)
    { }

  Date::Date(int m,int d):month(m),day(d)
    {year=2005;}

Date::Date(int m):month(m)
    {day=1;
```

```
    year=2005;
  }

Date::Date()
  {month=1;
    day=1;
    year=2005;
}

void Date::display()


int main()
{
  Date d1(10,13,2005);
  Date d2(12,30);
  Date d3(10);
  Date d4;
  d1.display();
  d2.display();
  d3.display();
  d4.display();
  return 0;
  }
```

3：

```
#include <iostream>
using namespace std;
class Date
  {public:
     Date(int=1,int=1,int=2005);
     void display();
   private:
     int month;
     int day;
     int year;
 };

Date::Date(int m,int d,int y):month(m),day(d),year(y)
  { }

void Date::display()


int main()
```

```
{
  Date d1(10,13,2005);
  Date d2(12,30);
  Date d3(10);
  Date d4;
  d1.display();
  d2.display();
  d3.display();
  d4.display();
  return 0;
}
```

4:

```
#include <iostream>
using namespace std;
class Student
  {public:
     Student(int n,float s):num(n),score(s){}
     void display();
   private:
     int num;
     float score;
  } ;

void Student::display()


int main()
{Student stud[5]={
    Student(101,78.5),Student(102,85.5),Student(103,98.5),
    Student(104,100.0),Student(105,95.5)};
  Student *p=stud;
  for(int i=0;i<=2;p=p+2,i++)
    p->display();
  return 0;
  }
```

5:

```
#include <iostream>
using namespace std;
class Student
  {public:
     Student(int n,float s):num(n),score(s){}
     int num;
```

```
    float score;
  };

void main()
{Student stud[5]={
   Student(101,78.5),Student(102,85.5),Student(103,98.5),
   Student(104,100.0),Student(105,95.5)};
  void max(Student* );
  Student *p=&stud[0];
  max(p);
  }

void max(Student *arr)
{float max_score=arr[0].score;
  int k=0;
  for(int i=1;i<5;i++)
     if(arr[i].score>max_score) {max_score=arr[i].score;k=i;}

}
```

6：

```
#include <iostream>
using namespace std;
class Student
{public:
     Student(int n,float s):num(n),score(s){}
     void change(int n,float s) {num=n;score=s;}

   private:
     int num;
     float score;
  };

int main()
{Student stud(101,78.5);
  stud.display();
  stud.change(101,80.5);
  stud.display();
  return 0;
}
```

7：解法一

```
#include <iostream>
using namespace std;
```

```
class Student
  {public:
     Student(int n,float s):num(n),score(s){}
     void change(int n,float s) {num=n;score=s;}

        //可改为
    private:
     int num;
     float score;
  };

int main()
{const Student
stud(101,78.5);
  stud.display();
  //stud.change(101,80.5);
  stud.display();
  return 0;
}
```

解法二：

```
#include <iostream>
using namespace std;
class Student
  {public:
     Student(int n,float s):num(n),score(s){}
     void change(int n,float s) const     {num=n;score=s;}

    private:
     mutable int num;
     mutable float score;
  };

int main()
{const Student stud(101,78.5);
  stud.display();
  stud.change(101,80.5);
  stud.display();
  return 0;
}
```

解法三：

```
#include <iostream>
using namespace std;
class Student
  {public:
     Student(int n,float s):num(n),score(s){}
     void change(int n,float s) {num=n;score=s;}

    private:
     int num;
     float score;
  };

int main()
{Student stud(101,78.5);
  Student *p=&stud;
  p->display();
  p->change(101,80.5);
  p->display();
  return 0;
}
```

8：

```
#include <iostream>
using namespace std;
class Student
  {public:
     Student(int n,float s):num(n),score(s){}
     void change(int n,float s) {num=n;score=s;}

    private:
     int num;
     float score;
  };

int main()
{Student stud(101,78.5);
  void fun(Student&);
  fun(stud);
  return 0;
}

void fun(Student &stu)
{stu.display();
stu.change(101,80.5);
```

```
  stu.display();
}
```

9：

```
#include <iostream>
using namespace std;
class Product
  {public:
     Product(int n,int q,float p):num(n),quantity(q),price(p){};
     void total();
     static float average();
     static void display();

   private:
     int num;
     int quantity;
     float price;
     static float discount;
     static float sum;
     static int n;
  };

void Product::total()
  {float rate=1.0;
    if(quantity>10) rate=0.98*rate;
    sum=sum+quantity*price*rate*(1-discount);
    n=n+quantity;
  }

void Product::display()
  {cout<<sum<<endl;
    cout<<average()<<endl;
  }

float Product::average()
  {return(sum/n);}


float Product::discount=0.05;
float Product::sum=0;
int Product::n=0;

int main()
  {
```

```
    Product Prod[3]={
        Product(101,5,23.5),Product(102,12,24.56),Product(103,100,21.5)
      };
    for(int i=0;i<3;i++)
        Prod[i].total();
    Product::display();
    return 0;
  }
```

10：

```
#include <iostream>
using namespace std;
class Date;
class Time
  {public:
      Time(int,int,int);
      friend void display(const Date &,const Time &);
    private:
      int hour;
      int minute;
      int sec;
  } ;

  Time::Time(int h,int m,int s)
  {hour=h;
    minute=m;
    sec=s;
  }

class Date
  {public:
  Date(int,int,int);
      friend void display(const Date &,const Time &);
    private:
      int month;
      int day;
      int year;
  } ;

Date::Date(int m,int d,int y)
  {month=m;
    day=d;
    year=y;
  }
```

```
void display(const Date &d,const Time &t)
  {


  }


int main()
{
  Time t1(10,13,56);
  Date d1(12,25,2004);
  display(d1,t1);
  return 0;
}
```

11：

```
#include <iostream>
using namespace std;
class Time;
class Date
  {public:
      Date(int,int,int);
      friend Time;
private:
      int month;
      int day;
      int year;
  };

Date::Date(int m,int d,int y):month(m),day(d),year(y){ }

class Time
  {public:
      Time(int,int,int);
      void display(const Date &);
    private:
      int hour;
      int minute;
      int sec;
  };

Time::Time(int h,int m,int s):hour(h),minute(m),sec(s){ }
```

```
void Time::display(const Date &d)
  {


  }


int main()
{
  Time t1(10,13,56);
  Date d1(12,25,2004);
  t1.display(d1);
  return 0;
}
```

12：

```
#include <iostream>
using namespace std;
template<class numtype>
class Compare
  {public:
     Compare(numtype a,numtype b);
     numtype max();
     numtype min();
   private:
     numtype x,y;
  };
template <class numtype>
Compare<numtype>::Compare(numtype a,numtype b)
   {x=a;y=b;}
template <class numtype>
numtype Compare<numtype>::max()
  {return (x>y)?x:y;}
template <class numtype>
numtype Compare<numtype>::min()
   {return (x<y)?x:y;}

int main()
{Compare<int> cmp1(3,7);


  Compare<float> cmp2(45.78,93.6);
```

```
  Compare<char> cmp3('a','A');


  return 0;
}
```

第四章

1：

```
#include <iostream>
using namespace std;
class Complex
  {public:
  Complex(){real=0;imag=0;}
     Complex(double r,double i){real=r;imag=i;}
     double get_real();
     double get_imag();
     void display();
   private:
     double real;
     double imag;
  };

double Complex::get_real()
   {return real;}

double Complex::get_imag()
   {return imag;}

   void Complex::display()


Complex operator + (Complex &c1,Complex &c2)
{
  return Complex(c1.get_real()+c2.get_real(),c1.get_imag()+c2.get_imag());
}

int main()
{Complex c1(3,4),c2(5,-10),c3;
  c3=c1+c2;

  c3.display();
  return 0;
```

```
}
2:
#include <iostream>
using namespace std;
class Complex
  {public:
  Complex(){real=0;imag=0;}
     Complex(double r,double i){real=r;imag=i;}
     Complex operator+(Complex &c2);
     Complex operator-(Complex &c2);
     Complex operator*(Complex &c2);
     Complex operator/(Complex &c2);
     void display();
   private:
     double real;
     double imag;
  };

Complex Complex::operator+(Complex &c2)
{Complex c;
  c.real=real+c2.real;
  c.imag=imag+c2.imag;
  return c;}

Complex Complex::operator-(Complex &c2)
{Complex c;
  c.real=real-c2.real;
  c.imag=imag-c2.imag;
  return c;}

Complex Complex::operator*(Complex &c2)
{Complex c;
  c.real=real*c2.real-imag*c2.imag;
  c.imag=imag*c2.real+real*c2.imag;
  return c;}

Complex Complex::operator/(Complex &c2)
{Complex c;
  c.real=(real*c2.real+imag*c2.imag)/(c2.real*c2.real+c2.imag*c2.imag);
  c.imag=(imag*c2.real-real*c2.imag)/(c2.real*c2.real+c2.imag*c2.imag);
  return c;}

void Complex::display()
```

```
int main()
{Complex c1(3,4),c2(5,-10),c3;
  c3=c1+c2;

  c3.display();
  c3=c1-c2;

  c3.display();
  c3=c1*c2;

  c3.display();
  c3=c1/c2;

  c3.display();
  return 0;
}
```

3：

```
#include <iostream>                //用 VC++时改为： #include <iostream.h>
using namespace std;               //用 VC++时为取消此行
class Complex
  {public:
  Complex(){real=0;imag=0;}
     Complex(double r,double i){real=r;imag=i;}
     Complex operator+(Complex &c2);
     Complex operator+(int &i);
     friend Complex operator+(int&,Complex &);
     void display();
   private:
     double real;
     double imag;
};

Complex Complex::operator+(Complex &c)
{return Complex(real+c.real,imag+c.imag);}

Complex Complex::operator+(int &i)
{return Complex(real+i,imag);}

void Complex::display()


Complex operator+(int &i,Complex &c)
{return Complex(i+c.real,c.imag);}
```

```
int main()
{Complex c1(3,4),c2(5,-10),c3;
  int i=5;
  c3=c1+c2;

  c3.display();
  c3=i+c1;

  c3.display();
  c3=c1+i;

  c3.display();
  return 0;
}
```

4:

```
#include <iostream>
using namespace std;
class Matrix                                          //定义 Matrix 类
  {public:
Matrix();                                             //默认构造函数
     friend Matrix operator+(Matrix &,Matrix &);      //重载运算符“+”
     void input();                                    //输入数据函数
     void display();                                  //输出数据函数
   private:
     int mat[2][3];
  };

Matrix::Matrix()                                      //定义构造函数
{for(int i=0;i<2;i++)
   for(int j=0;j<3;j++)
     mat[i][j]=0;
}

Matrix operator+(Matrix &a,Matrix &b)                 //定义重载运算符“+”函数
{Matrix c;
  for(int i=0;i<2;i++)
     for(int j=0;j<3;j++)
        {c.mat[i][j]=a.mat[i][j]+b.mat[i][j];}
  return c;
}
void Matrix::input()                                  //定义输入数据函数

  for(int i=0;i<2;i++)
```

```
    for(int j=0;j<3;j++)
      cin>>mat[i][j];
}

void Matrix::display()                                    //定义输出数据函数
{for (int i=0;i<2;i++)
{for(int j=0;j<3;j++)

       cout<<endl;}
}

int main()
{Matrix a,b,c;
    a.input();
    b.input();

    a.display();

    b.display();
 c=a+b;                                                   //用重载运算符“+”实现两个矩阵
相加

    c.display();
    return 0;
  }
  5:
  #include <iostream.h>
  //using namespace std;
  class Matrix
    {public:
       Matrix();
      friend Matrix operator+(Matrix &,Matrix &);
      friend ostream& operator<<(ostream&,Matrix&);
      friend istream& operator>>(istream&,Matrix&);
    private:
      int mat[2][3];
   };

Matrix::Matrix()
{for(int i=0;i<2;i++)
    for(int j=0;j<3;j++)
      mat[i][j]=0;
}
```

```
Matrix operator+(Matrix &a,Matrix &b)
{Matrix c;
  for(int i=0;i<2;i++)
      for(int j=0;j<3;j++)
          {c.mat[i][j]=a.mat[i][j]+b.mat[i][j];
          }
  return c;
}

istream& operator>>(istream &in,Matrix &m)

  for(int i=0;i<2;i++)
    for(int j=0;j<3;j++)
      in>>m.mat[i][j];
  return in;
}

ostream& operator<<(ostream &out,Matrix &m)
{for (int i=0;i<2;i++)
    {for(int j=0;j<3;j++)

        out<<endl;}
  return out;
}

int main()
{ Matrix a,b,c;
  cin>>a;
  cin>>b;

  c=a+b;

  return 0;
}
6：
#include <iostream>
using namespace std;
class Complex
  {public:
  Complex(){real=0;imag=0;}
      Complex(double r){real=r;imag=0;}
      Complex(double r,double i){real=r;imag=i;}
      operator double(){return real;}
```