清華大學
Tsinghua University

第11章 进阶内容导引

清华大学计算机系 韩文弢

教学内容

1. 文件
2. 模板
3. 标准模板库STL
4. 运算符重载

清華大學
Tsinghua University

1、文件的基本概念

- 什么是文件
  - **文件是一种抽象机制，它提供了一种把信息保存在磁盘等存储设备上，并且便于以后访问的方法。**
  - **在一个操作系统中，负责处理文件相关事宜的部分，称为文件系统。**

清華大學 Tsinghua University

文件的类型

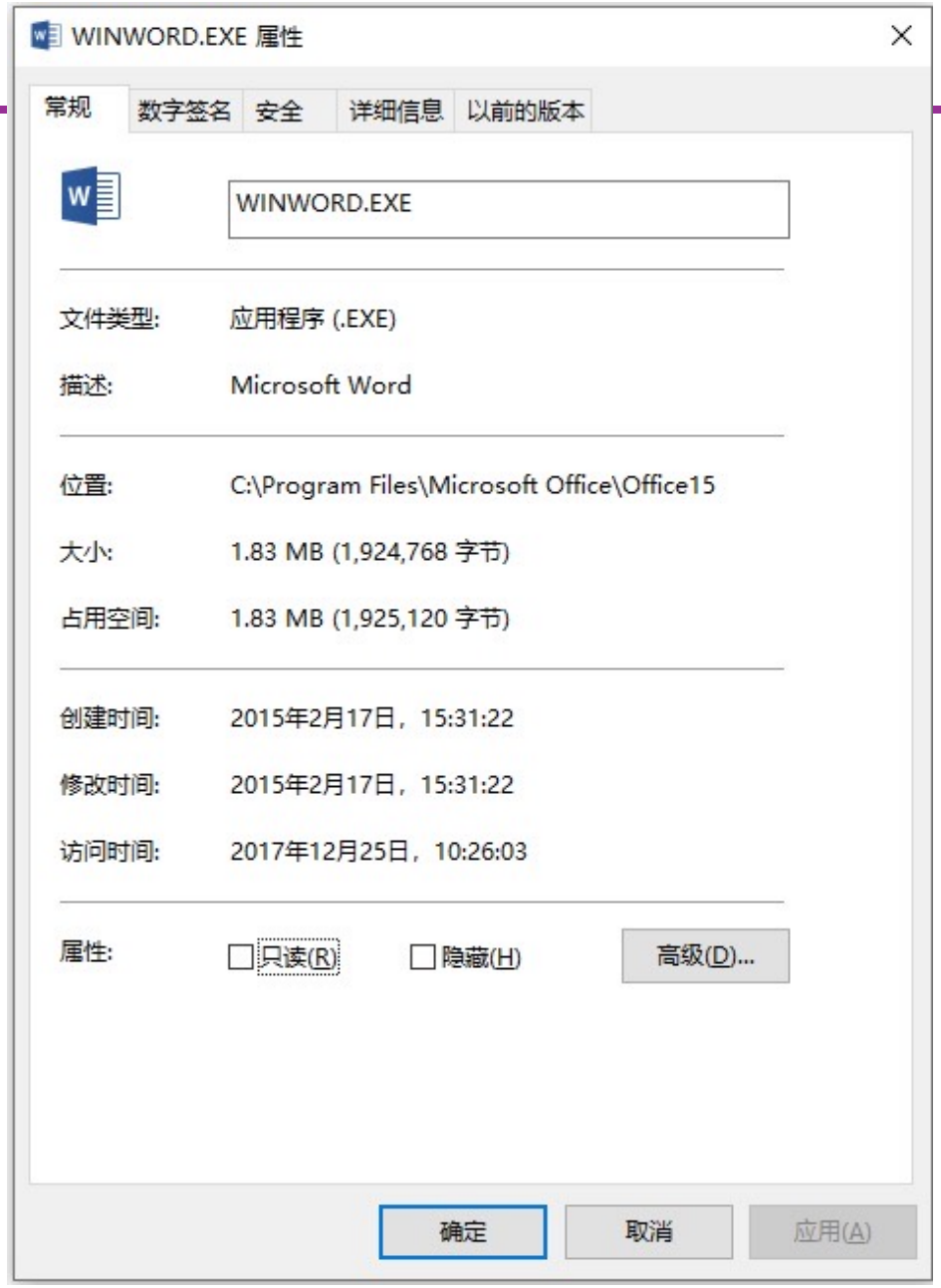
- 普通文件：包含用户信息的文件
  - ☺ 文本文件：由一行行文本组成
  - ☺ 二进制文件：非文本文件，通常具有某种内部的逻辑结构，为相关的应用程序所了解。
- 目录文件：管理文件系统结构的系统文件。

清華大學 Tsinghua University

文件的属性

- 每个文件都有一个名字和它所保存的信息，此外，OS还给每个文件附加了一些其他信息，这些信息称为文件的属性。
- 常见的一些文件属性：
  - ☺ 只读标志位：可读/写或只读
  - ☺ 隐藏标志位：普通文件或隐藏文件
  - ☺ 系统标志位：普通文件或系统文件
  - ☺ 创建时间、最后访问时间、最后修改时间
  - ☺ 文件长度
  - ☺ 文件的创建者、保护信息等

清華大學 Tsinghua University



文件的属性信息
存放在哪儿？

清華大學
Tsinghua University

文件的访问

- 访问文件的属性
  - ☺ 创建文件、删除文件、获取文件属性、设置文件属性、修改文件名、移动文件等
- 读写文件内容
  - ☺ 打开文件、关闭文件、读文件、写文件、添加、定位等

清華大學 Tsinghua University

目录

- **目录（directory）也称文件夹（folder），它是**一张表格，记录了在该目录下的每一个文件的文件名和其他的一些管理信息。
- 一般情况下，每个文件占用该表格的某一行，即一个目录项（该表格如何存放？）；
- 这张表格本身是以文件的形式存放在磁盘上；
- 在目录的管理上，也有相关的系统调用，如：
  - ☞创建目录、删除目录、修改目录名等

清華大學
Tsinghua University

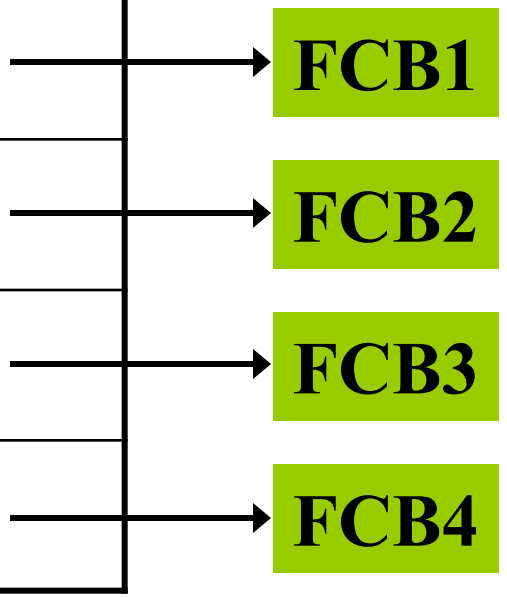
目录的实现

直接法

文件名	FCB
Games	FCB1
Mail	FCB2
News	FCB3
Work	FCB4

间接法

文件名	FCB索引
Games	
Mail	
News	
Work	



清華大學
Tsinghua University

2、访问文件属性

- C++访问文件属性
  - ☺ 主要通过函数来实现，没有专门的封装类，比较Java的File类
  - ☺ 在使用各个函数时，需将相应的头文件包含进来

清華大學
Tsinghua University

常用函数

函数名	头文件	功能描述

_findfirst();
_findnext();
_findclose();
	io.h	获取某个目录下所有文件的文件名

_access();	io.h	判断一个文件是否存在以及是否具有读写权限
_stat();	sys/stat.h	获取文件的长度、创建时间、上次访问时间和上次修改时间；判断当前文件是目录还是普通文件
rename();	cstdio或std::	修改一个文件的文件名
remove();	cstdio或std::	删除一个文件

清華大學
Tsinghua University

3、访问文件内容

- 文件处理与终端输入/输出类似
  - ☺ **头文件：#include <fstream>**
  - ☺ **ifstream类：从文件读入数据（文本或二进制）**
  - ☺ **ofstream类：向文件写入数据（文本或二进制）**

清華大學
Tsinghua University

4、读文本文件

```
#include <iostream>
#include <string>
#include <fstream>
using namespace std;
void main()
{
    ifstream infile("input.txt");
    int i;
    string s;
    infile >> i;  //类似于cin
    infile >> s;
    infile.close();
}
```


清華大學
Tsinghua University

符号单元

- 符号单元（token）：文件的内容被分隔为一个个的符号单元，中间用空白字符隔开
- 若一个文件包含如下内容，那么将被解释为

```
23    3.14
  "John  Smith"
```

Token	Type(s)
23	int, double, string
3.14	double, string
"John	string
Smith"	string

清華大學
Tsinghua University

输入光标

- 考虑一个文件weather.txt，其内容为：

```
16.2    23.5
    19.1 7.4   22.8


18.5   -1.8 14.9
```

- 整个文件被视为一个字符流

```
16.2    23.5\n    19.1 7.4   22.8\n\n18.5   -1.8 14.9\n
^
```

- 输入光标：文件的当前位置

清華大學
Tsinghua University

读入符号单元

- 读入数据：读入输入数据，将光标向后移动

```
16.2   23.5\n   19.1 7.4   22.8\n\n18.5   -1.8 14.9\n
 ^
```

```
double d; infile >> d;     // 16.2
```

```
16.2   23.5\n   19.1 7.4   22.8\n\n18.5   -1.8 14.9\n
     ^
```

```
string s; infile >> s;          // "23.5"
```

```
16.2   23.5\n   19.1 7.4   22.8\n\n18.5   -1.8 14.9\n
             ^
```


清華大學 Tsinghua University

5、写文本文件

```
#include <iostream>
#include <string>
#include <fstream>
using namespace std;
void main()
{
    ofstream outfile("output.txt");
    // ofstream outfile;
    // outfile.open("output.txt");
    string s("PI is: ");
    outfile << s; //类似于cout
    outfile << 3.14 << endl;
    outfile.close();
}
```


清華大學 Tsinghua University

读写文本文件的例子

- 访问文件**hours.txt**

```
123 Ben 12.5 8.1 7.6 3.2
456 Greg 4.0 11.6 6.5 2.7 12
789 Victoria 8.0 8.0 8.0 8.0 7.5
```

每人的工作天数不确定！

- 计算每个人的工作时间并写入文件

```
Ben (ID#123) worked 31.4 hours (7.85 hours/day)
Greg (ID#456) worked 36.8 hours (7.36 hours/day)
Victoria (ID#789) worked 39.5 hours (7.90 hours/day)
```


清華大學 Tsinghua University

基于行的处理方法

Method	Description
`std::getline()`	returns next entire line of input (from cursor to `\n`)

- **istringstream类可以分析一个字符串中的内容（头文件：sstream）**

  istringstream strstream(string line);

清華大學 Tsinghua University

参考程序

```
#include <iostream>
#include <string>
#include <fstream>
#include <ssream>
using namespace std;
void main()
{
    string line;
    ifstream input("hours.txt");
    ofstream output("output.txt");
    while(getline(input, line)){
        processEmployee(line, output);
    }
    input.close();
    output.close();
}
```

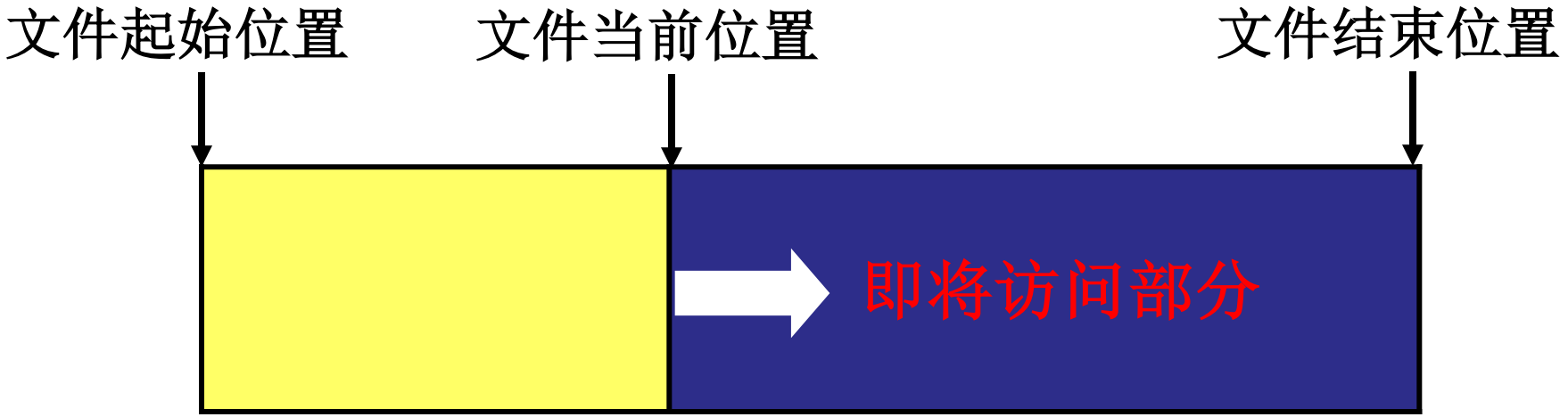

清華大學 Tsinghua University

```
void processEmployee(string &line, ofstream &output)
{
    int id, count = 0;
    string name;
    double sum = 0.0, hours, average;
    istringstream lineScan(line);
    lineScan >> id;             // e.g. 456
    lineScan >> name;           // e.g. "Greg"
    while (lineScan >> hours) {
        sum = sum + hours;
        count++;
    }
    average = sum / count;
    output << name << " (ID#" <<  id << ") worked "
       << sum << " hours (" << average
       << " hours/day)" << endl;
}
```


清華大學
Tsinghua University

6、读写二进制文件

- **随机存取**：根据所需访问的字节或记录在文件中的位置，将文件的读写指针直接移至该位置，然后进行存取。每一次存取操作都要指定该操作的起始位置。现代操作系统都提供这种方式。



清華大學
Tsinghua University

常用函数

函数名	功能描述
open();	将该流对象与某个特定的磁盘文件相关联
get();	从文件中读取一个字符
put();	向文件写入一个字符
read();	从文件的当前位置读取若干个字节的数据
write();	在文件的当前位置写入若干个字节的数据
seekg(); seekp();	用来设置文件读、写指针的当前位置
tellg(); tellp();	返回文件读、写指针的当前位置
close();	关闭文件

清華大學
Tsinghua University

字节流举例

有一首mp3歌曲的文件名为unknown.mp3，请编写一个程序，显示其歌名和歌手

参考程序

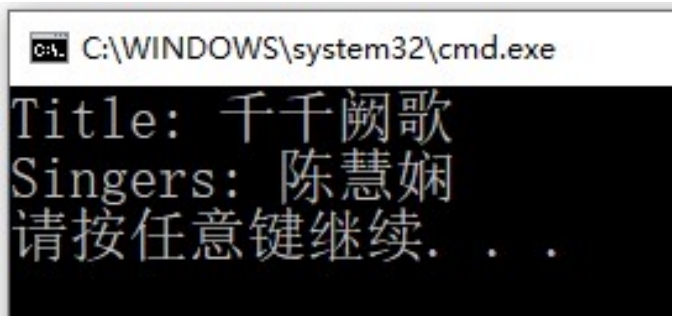
```
#include <iostream>
#include <fstream>
#include <sys/stat.h>
using namespace std;
void main()
{
    int fileLength;
    char tagBuf[128], *pSongName, *pArtist;
    struct _stat info;

    ifstream input("unknown.mp3", ios_base::binary);
    if(!input) return ;
```


清華大學 Tsinghua University

参考程序（2）

```
    _stat("unknown.mp3", &info);
    fileLength = info.st_size;
    input.seekg(fileLength-128);
    input.read(tagBuf, 128);
    input.close();
    pSongName = &tagBuf[3];
    pArtist = &tagBuf[33];
    cout << "Title: " << pSongName << endl;
    cout << "Singers: " <<  pArtist << endl;
}
```



清華大學
Tsinghua University

教学内容

1. 文件
2. 模板
3. 标准模板库STL
4. 运算符重载

清華大學 Tsinghua University

1、函数模板

如何实现用同一个函数名处理不同类型的数据？

函数重载！

重载法求绝对值

```
int abs(int x)
{
    if(x >= 0) return x;
    else return -x;
}
```

```
double abs(double x)
{
    if(x >= 0) return x;
    else return -x;
}
```

. . . . . .

清華大學
Tsinghua University

函数模板法

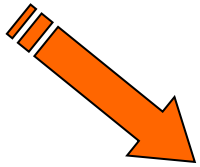
```
template <typename T> //名字不一定为T
// template <class T>
T abs(T x) //类型参数，可以是任何可能的类型
{
    T result;
    if(x >= 0) result = x;
    else result = -x;
    return result;
}
...
cout << abs<double>(-3.14); //函数调用
```


清華大學
Tsinghua University

实现过程

```
cout << abs<double>(-3.14);
```

```
template <typename T>
T abs(T x)
{
    T result;
    if(x >= 0) result = x;
    else result = -x;
    return result;
}
```



```
double abs(double x)
{
    double result;
    if(x >= 0) result = x;
    else result = -x;
    return result;
}
```

```
cout << abs<int>(-3);
```


清華大學
Tsinghua University

2、类模板

类是成员变量和成员函数的封装，因此，可以把函数模板推广到类

类模板举例

```
#include <iostream>
using namespace std;
template <typename T>
class Point {
private:
    T x, y;
public:
    Point(T u, T v) : x(u), y(v) {}
    T getX() { return x; }
    T getY() { return y; }
};
```


清華大學
Tsinghua University

类模板举例(2)

```
int main
{
    Point<double> dpoint(2.5, 3.5);
    cout << dpoint.getX() << ", "
         << dpoint.getY() << endl;
    return 0;
}
```

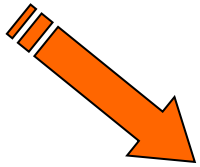
```
// 成员函数声明与实现分离
template <typename T>
T Point<T>::getX() { return x; }
```


清華大學
Tsinghua University

实现过程

```
Point<double> dpoint(2.5, 3.5);
```

```
template <typename T>
class Point {
private:
    T x, y;
public:
    Point(T u, T v)
        : x(u), y(v) {}
    T getX() { return x; }
    T getY() { return y; }
};
```



```
class Point {
private:
    double x, y;
public:
    Point(double u,double v)
        : x(u), y(v) {}
    double getX(){return x;}
    double getY(){return y;}
};
```


清華大學 Tsinghua University

教学内容

1. 文件
2. 模板
3. 标准模板库STL
4. 运算符重载

清華大學 Tsinghua University

标准模板库

- 标准模板库STL
  - Standard Template Library
  - C++标准库的一个组成部分
  - 包含了许多有用的数据结构类和算法
  - 用模板（泛型）来实现，通用性好

清華大學
Tsinghua University

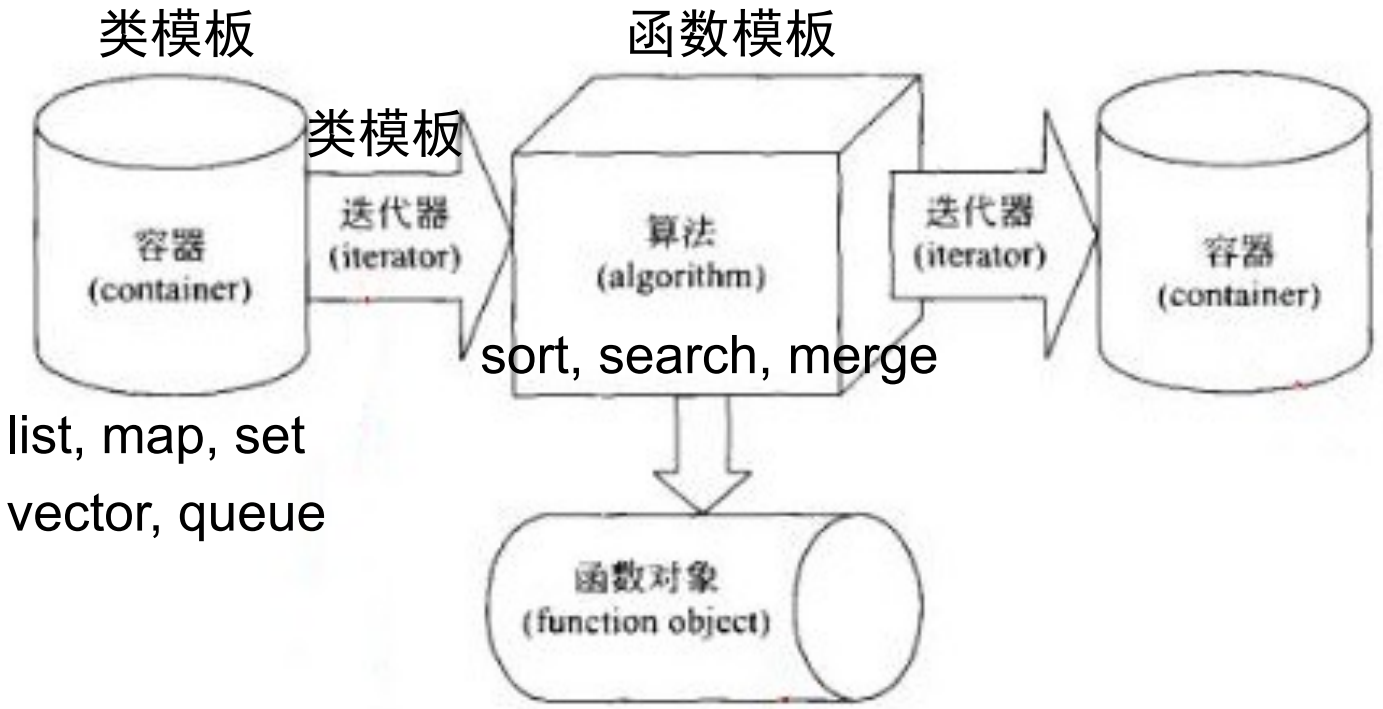
STL的基本组件

- ☺ 排序对象：任何东西（整数、实数、对象）
- ☺ 访问方式：统一方式（首个、下一个）
- ☺ 排序方法：任何算法（冒泡、选择、快速）
- ☺ 比较方式：任何方式（大小、长度）

容器 **container**	迭代器 **iterator**
算法 **algorithm**	函数对象 **function**

清華大學
Tsinghua University

组件间的关系



list, map, set

vector, queue

清華大學
Tsinghua University

一个例子

```
#include <iostream>
#include <set>
#include <algorithm>
using namespace std;
int main()
{
    set<int> iset;
    iset.insert(5);
    iset.insert(9);
    iset.insert(1);
    iset.insert(8);
    iset.insert(3);
    cout << "iset contains:";
    set<int>::iterator it;
```


清華大學 Tsinghua University

一个例子(2)

```
    for(it=iset.begin();it!=iset.end();it++)
        cout << " " << *it;
    cout << endl;
    int searchFor;
    cin >> searchFor;
    if(binary_search(iset.begin(),
                     iset.end(), searchFor))
        cout << "Found " << searchFor;
    else
        cout << "Did not find " <<searchFor;

    return 0;
}
```


清華大學
Tsinghua University

教学内容

1. 文件
2. 模板
3. 标准模板库STL
4. 运算符重载

清華大學
Tsinghua University

对自定义类型进行运算

```
struct Complex{
    double real;
    double imag;
};
int main()
{
    struct Complex x = {2.5, 3.5};
    struct Complex y = {1, 1.5};
    struct Complex z;
    z = x + y;    // 编译错误!
    return 0;
}
```


清華大學
Tsinghua University

+运算符重载

```
struct Complex operator+(struct Complex x,
                         struct Complex y)
{
    struct Complex temp = {0, 0};
    temp.real = x.real + y.real;
    temp.imag = x.imag + y.imag;

    return temp;
}
```

<<运算符重载

```
ostream& operator<<(ostream &o,
                    struct Complex &x)
{
    o << x.real << "+" << x.imag << "i";
    return o;
}
int main()
{
    struct Complex x = {2.5, 3.5};
    struct Complex y = {1, 1.5};
    struct Complex z;
    z = x + y;
    cout << z << endl;
    return 0;
}
```


清華大學 Tsinghua University

允许重载的运算符

```
+     -     *     /     +=    -=    *=    /=
%     %=    ++    --    =     ==    <     >
<=    >=    !     !=    &&    ||    <<    >>
<<=   >>=   &     ^     |     &=    ^=    |=
~     []    ()    ,     ->*   ->    new   new[]
delete  delete[]
```


清華大學
Tsinghua University

本讲小结

- 文件操作
- 模板和标准库
- 运算符重载

清華大學
Tsinghua University