



# 最简单的C++程序

北京大学信息科学技术学院

Courtesy 李戈 《计算概论》



北京大学

```
#include<iostream>
using namespace std;
int main( )
{
    int number[45] = {78, 56, 69, 31, 36, 67, 31, 47, 69, 34, 45, 74, 61,
        82, 43, 41, 76, 79, 81, 66, 54, 50, 76, 51, 53, 28, 74, 39, 45, 61, 52, 41,
        43, 75, 78, 84, 72, 51, 43, 64, 75, 81, 69, 55, 74};
    int max = 0;
    int i = 0;
    for(i = 0; i < 45; i++)
    {
        if(number[i] > max)
            max = number[i];
    }
    cout<<"The Maximal Number is:"<<max;
    return 0;
}
```

### **问题1：**

**是不是 无论我们在程序里写什么“单词”，计算机都能明白？**

### **问题2：**

**是不是 无论我们在程序里写什么“数”和“计算符号”，计算机都能明白？**

### **问题3：**

**世界上可能要用“程序来表达的逻辑”纷繁复杂，程序设计语言里面得有多少种“句式”才能够用？**

## 问题1：

是不是 无论我们在程序里写什么“单词”，计算机都能明白？

◆ NO！

◆ 编程语言定义了一些有特定含义的“关键字”，计算机“只能明白”这些“词”的含义。



# 计算机能够认识的单词

auto	break	case	char	const
continue	default	do	double	else
enum	extern	float	for	goto
if	int	long	register	return
short	signed	sizeof	static	struct
switch	typedef	unsigned	union	void
volatile	while	bool	catch	class



```
#include<iostream>
using namespace std;
int main()
{
    enum day{Mon, Tue, Wed, Thu, Fri, Sat, Sun};
    day workDay;
    double times, wages, hourlyRate, hours;
    cout<<"Enter the hourly wages rate."<<endl;
    cin>>hourlyRate;
    cout<<"Enter hours worked daily\n";
    for (workDay=mon; workDay<=sun; workDay++)
    {
        cin>>hours; //输入周一到周日的工作时间 ;
        switch(workDay)
        {
            case sat: times=1.5*hours; break;
            case sun: times=2.0*hours; break;
            default: times=hours; }
        wages = wages + times*hourlyRate;
    }
    cout<<"The wages for the week are "<<wages;
    return 0 ;
}
```

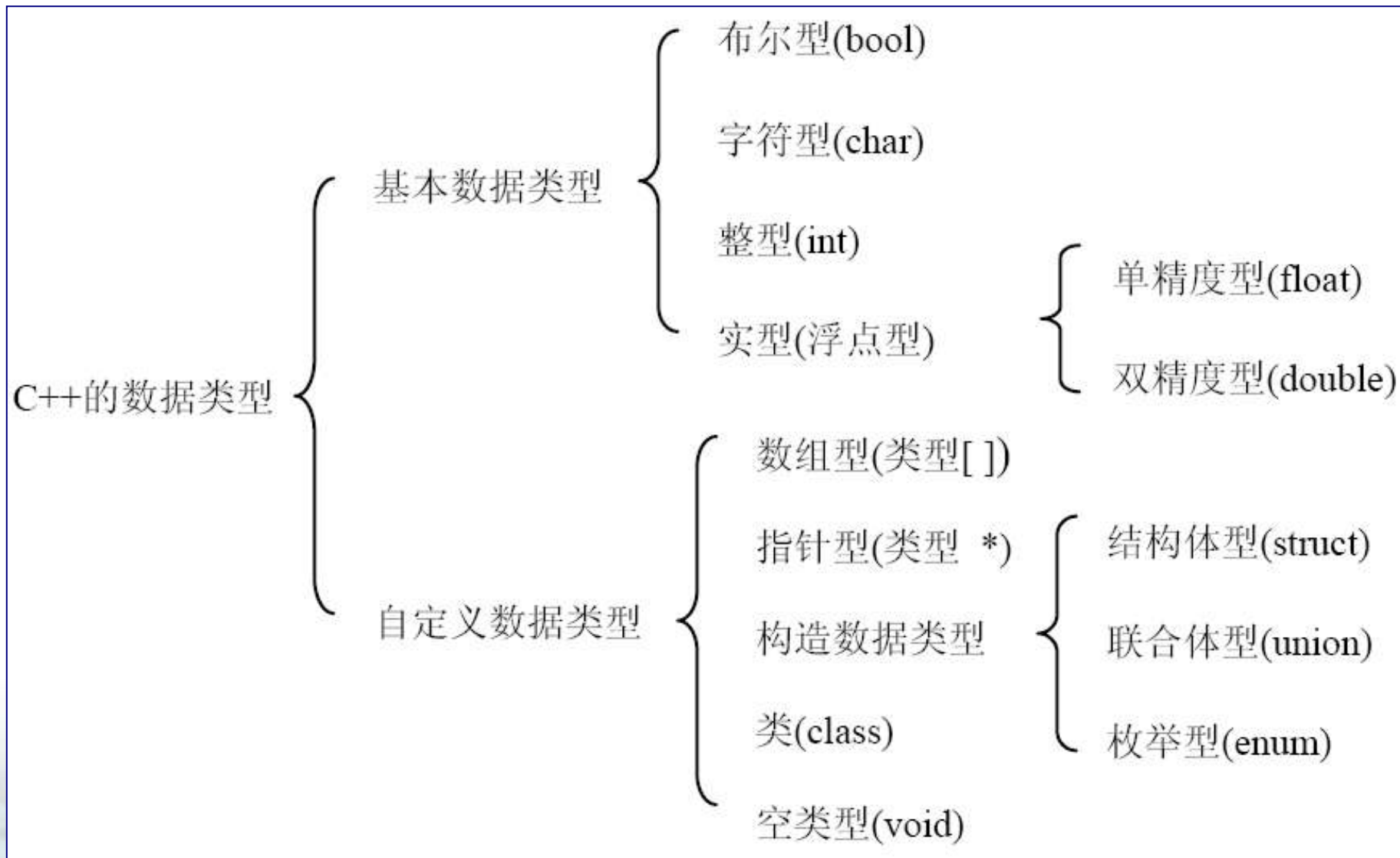
## 问题2：

是不是 无论我们在程序里写什么“数”和“计算符号”，计算机都能明白？

◆ NO！

◆ 计算机只能“看懂”某些类型的数据，这些“数据的类型”和相应的“操作符号”也是定义好的。

# 计算机能够看懂的数据的类型





# 计算机能够理解的运算的种类

## ■ C++语言的运算符

- ◆ 求字节数运算符: `sizeof`
- ◆ 下标运算符 `[]`
- ◆ 赋值运算符 `=`
- ◆ 算术运算符 `+ - * / %`
- ◆ 关系运算符 `< > == >= <= !=`
- ◆ 逻辑运算符 `! && ||`
- ◆ 条件运算符 `? :`
- ◆ 逗号运算符 `,`
- ◆ 位运算符 `>> ~ | ^ &`
- ◆ 指针运算符 `* , &`
- ◆ 强制类型转换运算符: `(类型)`
- ◆ 分量运算符 `. →`



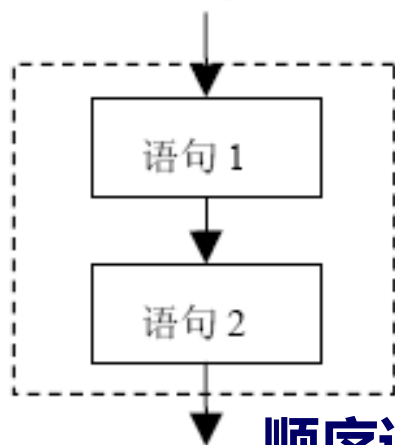
### 问题3：

**世界上可能要用“程序来表达的逻辑”纷繁复杂，程序设计语言里面得有多少种“句式”才能够用？**

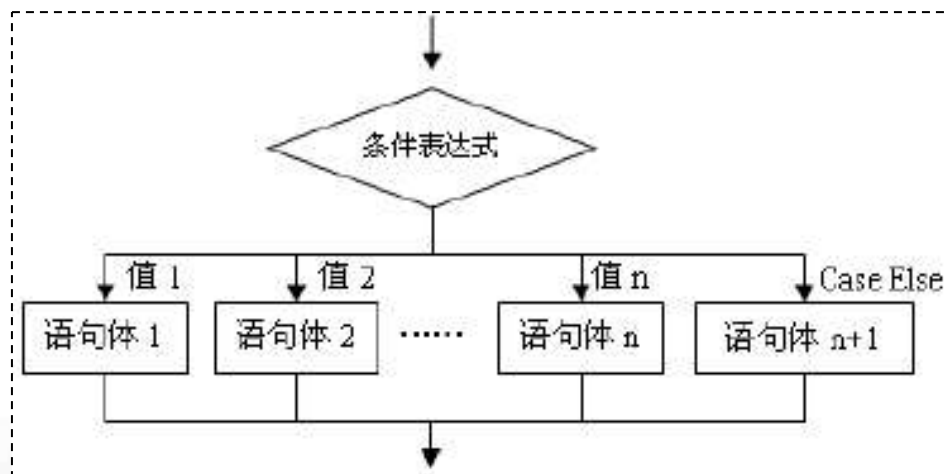
**◆ 不多！三种而已！**

- C. Bohm & G. Jacopini, "Flow Diagrams, Turing Machines and Languages with Only Two Formation Rules," Communications of the ACM, vol9(5) May 1966, pp 366-371.

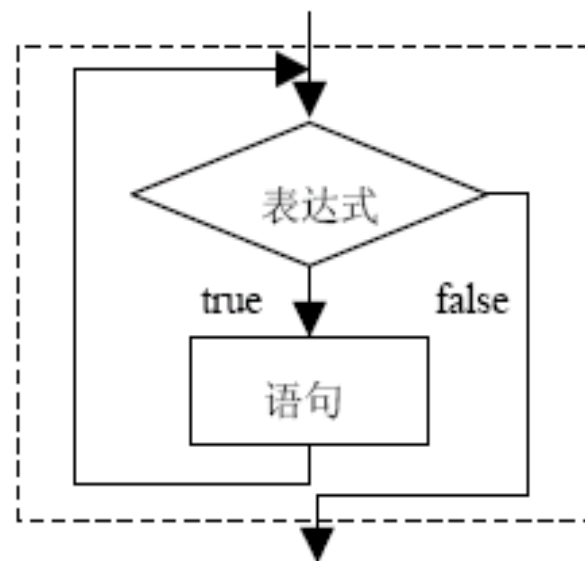
# 如何表达纷繁复杂的计算逻辑？



**顺序语句**



**分支语句**



**循环语句**



北京大学

# 我们要学的编程语言

**30几个关键字**

**十几种基本数据类型 + 30几个运算符**

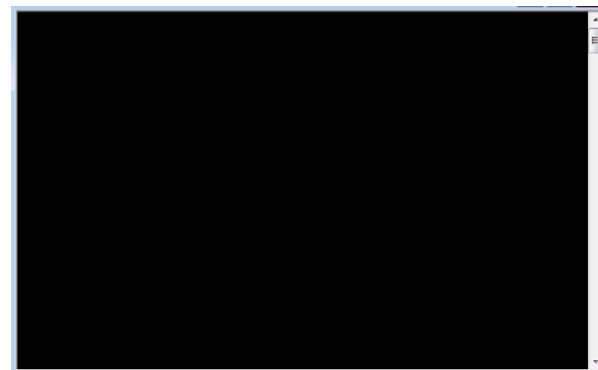
**三种基本的逻辑语句**

# 最简单的程序

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

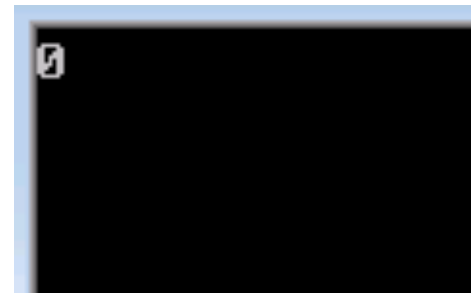
    return 0;

}
```



# 很简单的程序 (1)

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



# 变量的定义和使用

## ■ 变量的定义及初始化

◆ 变量类型 变量名称 ;

◆ int i ;

◆ float result;

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

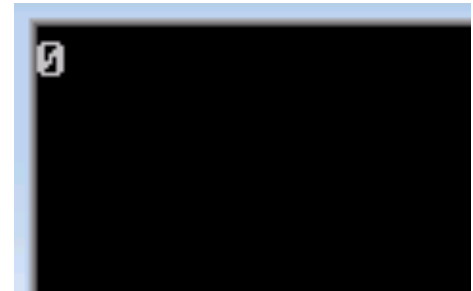
## 变量的初始化 :

◆ int i = 0 ;

相当于 :

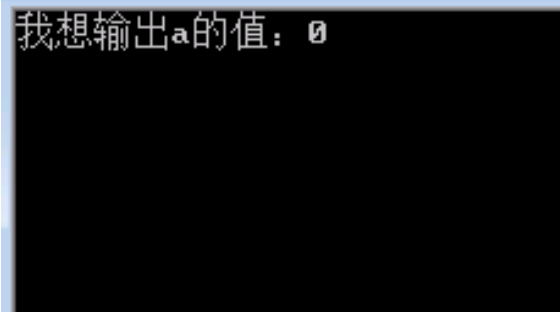
◆ int i;

◆ i = 0 ;



## 很简单的程序 (2)

```
#include <iostream>
using namespace std;
int main()
{
    int a = 0;
    cout<<"我想输出a的值: "<<a<<endl;
    return 0;
}
```



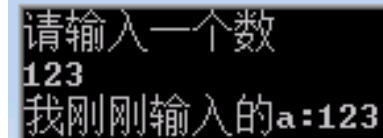
我想输出a的值: 0





## 很简单的程序 (3)

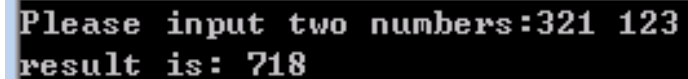
```
#include <iostream>
using namespace std;
int main()
{
    int a = 0;
    cout << "请输入一个数" << endl;
    cin >> a;
    cout << "我刚刚输入的a:" << a << endl;
    return 0;
}
```



请输入一个数  
123  
我刚刚输入的a:123

# 简单的程序 (1) —— 顺序结构

```
#include <iostream>
using namespace std;
int main()
{
    int a = 0, b = 0, result = 0;
    cout << "Please input two numbers:";
    cin >> a >> b;
    result = 3*a-2*b+1;
    cout << "result is : " << result << endl;
    return 0;
}
```

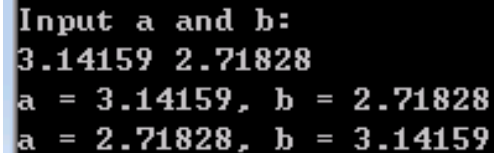


```
Please input two numbers:321 123
result is: 718
```



## 简单的程序（2）——顺序结构

```
#include<iostream>
using namespace std;
int main()
{
    float a = 0, b = 0, temp = 0;
    cout << "Input a and b:"<<endl;
    cin >> a >> b;
    cout << "a = "<< a << ", b = " << b<<endl;
    temp = a; a = b; b = temp;
    cout << "a = " << a << ", b = " << b<<endl;
    return 0;
}
```

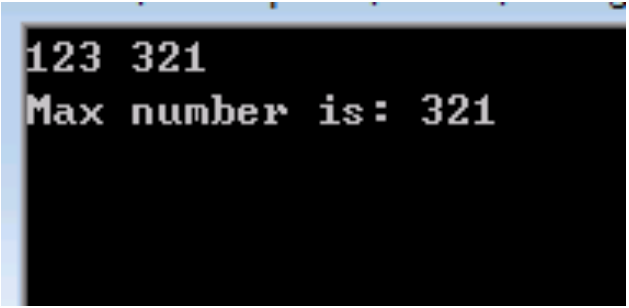


```
Input a and b:
3.14159 2.71828
a = 3.14159, b = 2.71828
a = 2.71828, b = 3.14159
```



## 简单的程序 (3) —— 分支结构

```
#include<iostream>
using namespace std;
int main()
{
    int x = 0, y = 0;
    cin >> x >> y;
    if (x > y)
        cout << "Max number is: " << x << endl;
    else
        cout << "Max number is: " << y << endl;
    return 0;
}
```



```
123 321
Max number is: 321
```



## 回顾——if 语句

### ◆形式一

if (条件表达式)

单一语句 ;

else

单一语句 ;

最常用的条件表达式 :

$A > B, A \geq B$

$A < B, A \leq B$

$A == B$

### ◆形式二

if (条件表达式)

{

语句组 ;

}

else

{

语句组 ;

}



北京大学

## 简单的程序（4）——分支结构

```
#include<iostream>
using namespace std;
int main()
{
    char a = ' ';
    cout << "猜猜我是哪个字母:" << endl;
    cin >> a;
    if (a != 'G')
        cout << "你猜错了！" << endl;
    else if (a == 'G')
        cout << "被你猜中了！" << endl;
    return 0;
}
```

# 字符型数据

## ■ 字符型变量

◆ `char a = 'a';`

◆ 用单引号括起来的单个字符。如 `'a'`, `'x'`, `'D'`, `'#'`

```
#include <iostream>
using namespace std;
int main( )
{
    char a = 'a';
    cout<<a<<endl;
    cout<<"This is the first line ! \n";
    cout<<'a'<<endl;
    return 0;
}
```



# 简单的程序（4）——分支结构

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

```
    char a = ' ';
    cout << "猜猜我是哪个字母:" << endl;
    cin >> a;
    if (a != 'G')
        cout << "你猜错了！" << endl;
    else if (a == 'G')
        cout << "被你猜中了！" << endl;
    return 0;
}
```

猜猜我是哪个字母：  
a  
你猜错了！

猜猜我是哪个字母：  
G  
被你猜中了！



## 简单的程序 (5) —— 循环结构

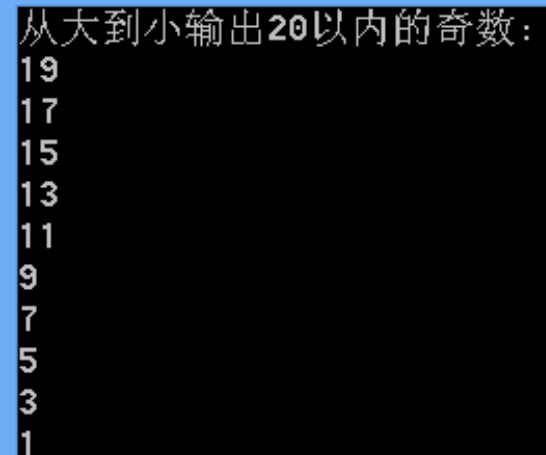
```
#include<iostream>
using namespace std;
int main()
{
    int i = 0;
    cout << "20以内的奇数:" << endl;
    for (i = 0; i < 20; i++)
    {
        if (i % 2 != 0)
            cout << i << endl;
    }
    return 0;
}
```

20以内的奇数:

1  
3  
5  
7  
9  
11  
13  
15  
17  
19

## 程序 (6) —— 循环结构

```
#include<iostream>
using namespace std;
int main()
{
    int i = 0;
    cout << "从大到小输出20以内的奇数:" << endl;
    for (i = 20; i >= 0; i--)
    {
        if (i % 2 != 0)
            cout << i << endl;
    }
    return 0;
}
```



从大到小输出20以内的奇数:  
19  
17  
15  
13  
11  
9  
7  
5  
3  
1

# 简单的程序 (7) —— 数组

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

```
    int i = 0;
```

```
    char a[10] = { 'a', 'b', 'c', 'd', 'e', 'f', 'g', 'h', 'i', 'j' };
```

```
    cout << “字母表中序号为奇数的前五个字母:” << endl;
```

```
    for (i = 0; i < 10; i=i+2)
```

```
    {
```

```
        cout << a[i] << endl;
```

```
    }
```

```
    return 0;
```

```
}
```

输出字母表中序号为奇数的五个字母：

a  
c  
e  
g  
i



## ■ 上一页程序为什么会出错



北京大学



# 数字化原理-英文字符编码： ASCII码

- **ASCII码**

- 美国信息交换标准码 (American Standard Code for Information Interchange)
- 国际上使用最广泛的字符编码

- **ASCII码的编码规则**

- 每个字符用7位二进制数( $d_6d_5d_4d_3d_2d_1d_0$ )来表示；  
可表示128个字符 (7位二进制共有128种状态,  $2^7 = 128$ )

- **ASCII码在计算机中的存储**

- 在计算机中，每个字符的ASCII码用一个字节存
- 字节的最高位 $d_7$ 为校验位（奇偶校验），通常用0填充；  
字节的后7位为实际编码值



北京大学



# 数字化原理-英文字符编码： ASCII码

L \ H	0000	0001	0010	0011	0100	0101	0110	0111
0000	NUL	DLE	SP	0	@	P	,	p
0001	SOH	DC1	!	1	A	Q	a	q
0010	STX	DC2	"	2			.	.
0011	ETX	DC3	#	3				
0100	EOT	DC4	\$	4	D	I	d	t
0101	ENQ	NAK	%	5	E	U	e	u
0110	ACK	SYN	&	6	F	V	f	v
0111	BEL	ETB	,	7	G	W	g	w
1000	BS	CAN	)	8	H	X	h	x
1001	HT	EM	(	9	I	Y	i	y
1010	LF	SUB	*	:	J	Z	j	z
1011	VT	ESC	+	;	K	[	k	{
1100	FF	FS	,	<	L	\	l	
1101	CR	GS	-	=	M	]	m	}
1110	SO	RS	.	>	N	^	n	~
1111	SI	US	/	?	O	_	o	DEL

A:  $(01000001)_2 / (65)_{10}$

x: ?

ASCII  
码



北京大学



# 数字化原理-汉字编码

- 常用汉字数量：3000~5000个
  - 无法用一个字节来编码
- 国标码《通用汉字字符集（基本集）及其交换码标准》
  - 共收集了7445个字符
  - 用两个字节来编码一个汉字；每个字节的最高位为0
- 国标码在计算机中的存储：汉字内码
  - 把国标码两个字节的最高位变为1（避免和ASCII码发生冲突）

汉字 啊

国标码：00110000 00100001

内码：10110000 10100001



北京大学



# 数 组

## ■ 定义其他类型的数组并附初始值

- ◆ `int a[26];`
- ◆ `float a[26];`
- ◆ `bool a[26];`
- ◆ `char a[26] = {'a', 'b', 'c', ..., 'z'};`

## ■ 数组的使用

- ◆ 要使用第 $n$ 个数：`a[n-1];`    `a[n-1] = 1+2;`
- ◆ 注意：对数组`a[n]`，只能使用`a[0] – a[n-1];`  
`a[n]`不能用；







```
char a[10] = { 'a', 'b', 'c', 'd', 'e', 'f', 'g', 'h', 'i', 'j' };
```

a	b	c	d	e	f	g	h	i	j
---	---	---	---	---	---	---	---	---	---

a[0]	a[1]	a[2]	a[3]	a[4]	a[5]	a[6]	a[7]	a[8]	a[9]
------	------	------	------	------	------	------	------	------	------

1	2	3	4	5	6	7	8	9	10
---	---	---	---	---	---	---	---	---	----

```
int a[10] = { 1, 2, 3, 4, 5, 6, 7, 8, 9, 10 };
```



# 数组的使用

```
#include<iostream>
using namespace std;
int main( )
{
    int number[45] = {78, 56, 69, 31, 36, 67, 31, 47, 69, 34, 45, 74, 61, 82,
43, 41, 76, 79, 81, 66, 54, 50, 76, 51, 53, 28, 74, 39, 45, 61, 52, 41, 43, 75,
78, 84, 72, 51, 43, 64, 75, 81, 69, 55, 74};
    int max = 0;
    int i = 0;
    for(i = 0; i < 45; i++)
    {
        if(number[i] > max)
            max = number[i];
    }
    cout<<"The Maximal Number is:"<<max;
    return 0;
}
```



# 程序 (8) —— 综合

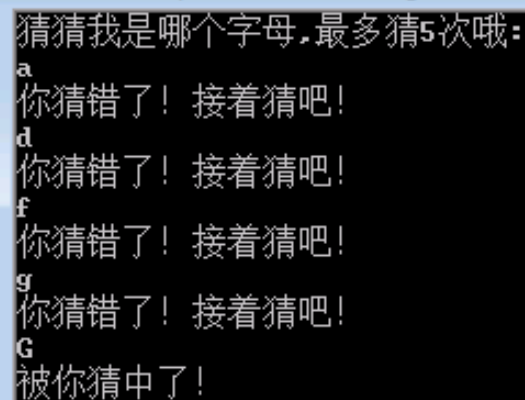
```
#include<iostream>
using namespace std;
int main()
{
    char a = ' ';
    cout << "猜猜我是哪个字母,最多猜5次哦:" << endl;
    int i = 0;
    for (i = 0; i < 5; i++)
    {
        cin >> a;
        if (a != 'G')
            cout << "你猜错了! 接着猜吧! " << endl;
        else
        {
            cout << "被你猜中了! " << endl;
            break;
        }
    }
    return 0;
}
```

# 程序 (8) —— 综合

```
#include<iostream>
using namespace std;
int main()
{
    char a = ' '; //用于存放用户输入的字母
    cout << "猜猜我是哪个字母,最多猜5次哦:" << endl;
    int i = 0; //用于记录猜过多少次了
    for (i = 0; i < 5; i++)
    {
        cin >> a;
        if (a == 'G') //如果猜中
        {
            cout << "被你猜中了! " << endl;
            break; //终止循环
        }
        else //如果没有猜中
            cout << "你猜错了! 接着猜吧! " << endl;
    }
    return 0;
}
```

# 程序 (8) —— 综合

```
#include<iostream>
using namespace std;
int main()
{
    char a = ' '; //用于存放用户输入的字母
    cout << "猜猜我是哪个字母,最多猜5次哦!" << endl;
    int i = 0; //用于记录猜过多少次了
    for (i = 0; i < 5; i++)
    {
        cin >> a;
        if (a == 'G') //如果猜中
        {
            cout << "被你猜中了! " << endl;
            break; //终止循环
        }
        else //如果没有猜中
            cout << "你猜错了! 接着猜吧! " << endl;
    }
    return 0;
}
```



```
猜猜我是哪个字母,最多猜5次哦:
a
你猜错了! 接着猜吧!
d
你猜错了! 接着猜吧!
f
你猜错了! 接着猜吧!
g
你猜错了! 接着猜吧!
G
被你猜中了!
```

## 程序 (9) —— 综合

```
#include<iostream>
using namespace std;
int main()
{
    char input = ' ';    //用于存放输入的字母
    char letter[5] = { 'a', 'e', 'i', 'o', 'u' };
    cout<<"我可能是哪些字母？猜5次，看你最多猜对几个？"<< endl;
    int count = 0;        //用于记录猜对了几个
    int temp = 0;         //用于临时记录count的值
    int i = 0, j = 0;
    for (i = 0; i < 5; i++) //循环猜三次
    {
        cin >> input;
        temp = count;
        for (j = 0; j < 5; j++) //逐个对比输入字母与已有字母
        {
            if (input == letter[j])
            {
                cout<< "你猜对了！" << endl;
                count++; //如果猜对了，计数器加1
                break;
            }
        }
        if (count == temp) //比完5次但count没增加
            cout<< "没有猜对！" << endl;
    }
    cout<< "你一共猜对了" << count << "个" << endl;
    return 0;
}
```

## 程序 (9) —— 综合

程序的含义：

藏起五个字母让你来猜，最多  
猜五次，看你最多猜对几个

```
#include<iostream>
using namespace std;
int main()
{
    char input = ' ';    //用于存放输入的字母
    char letter[5] = { 'a', 'e', 'i', 'o', 'u' };
    cout<<"我可能是哪些字母？猜5次，看你最多
    int count = 0;        //用于记录猜对了几个
    int temp = 0;        //用于临时记录count的值
    int i = 0, j = 0;
    for (i = 0; i < 3; i++) //循环猜三次
    {
        cin >> input;
        temp = count;
        for (j = 0; j < 5; j++) //逐个对比输入字母与已有字母
        {
            if (input == letter[j])
            {
                cout<<"你猜对
                count++; //如果
                break;
            }
        }
        if (count == temp) //比完5次但count
        cout<<"没有猜对！" << endl;
    }
    cout<<"你一共猜对了" << count <<"个" << endl;
    return 0;
}
```

```
我可能是哪些字母？猜3次，看你最多猜对几个？
a
你猜对了！
f
没有猜对！
e
你猜对了！
你一共猜对了2个
```



# 关于C++程序的物理结构

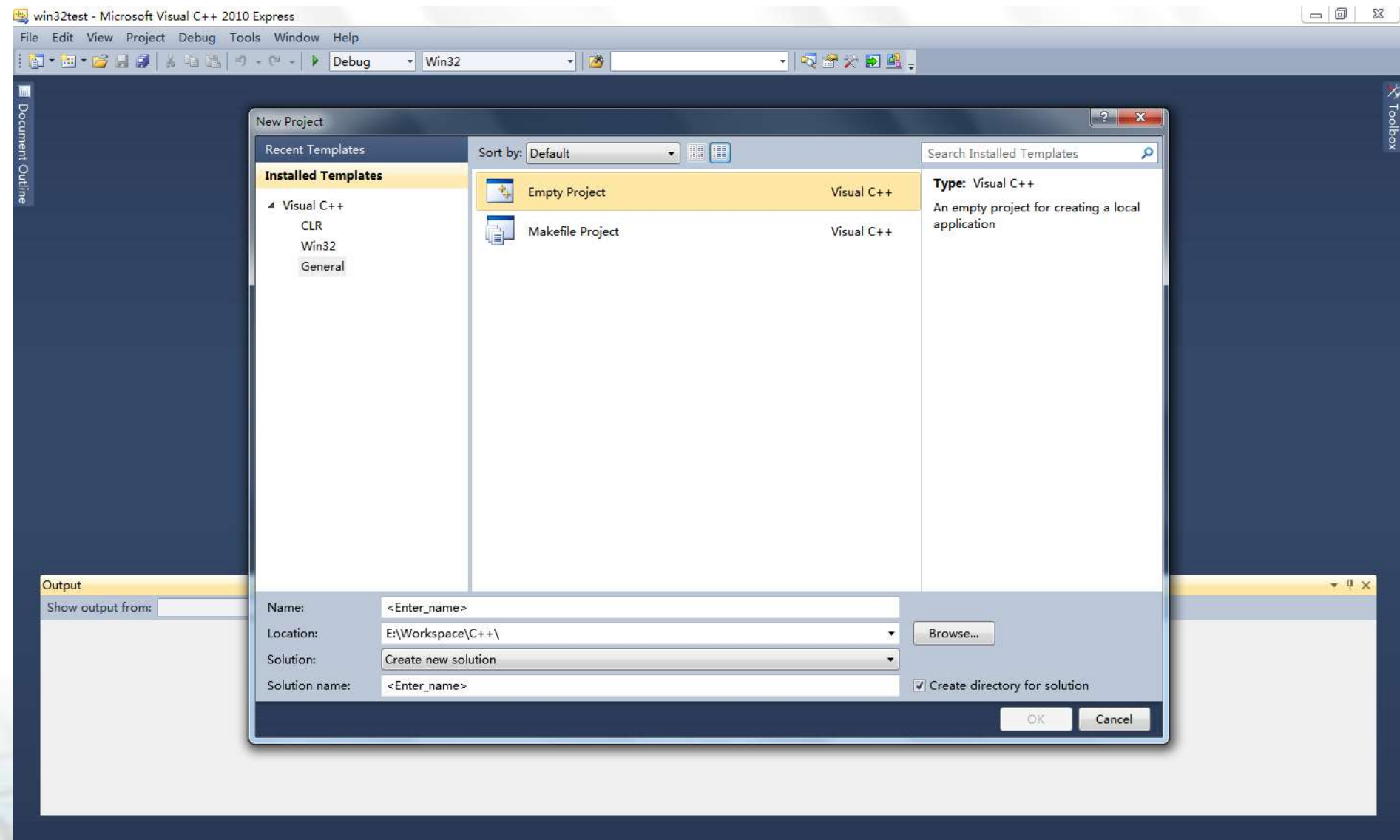
## ■ 工程 ( Project )

- ◆ 每个C++程序必须保存在以.cpp为后缀的文件中；
- ◆ 每个.cpp需要 “生活在” 一个工程 ( Project ) 中；
  - 每个工程中可以包含多个.cpp文件；
  - 每个.cpp文件中可以包含多个 函数；
- ◆ 每个工程中，只能包含一个 `int main()` 函数；
  - `main()` 函数是工程执行的入口；
- ◆ 可以引用函数库里的函数，但需要 `#include`

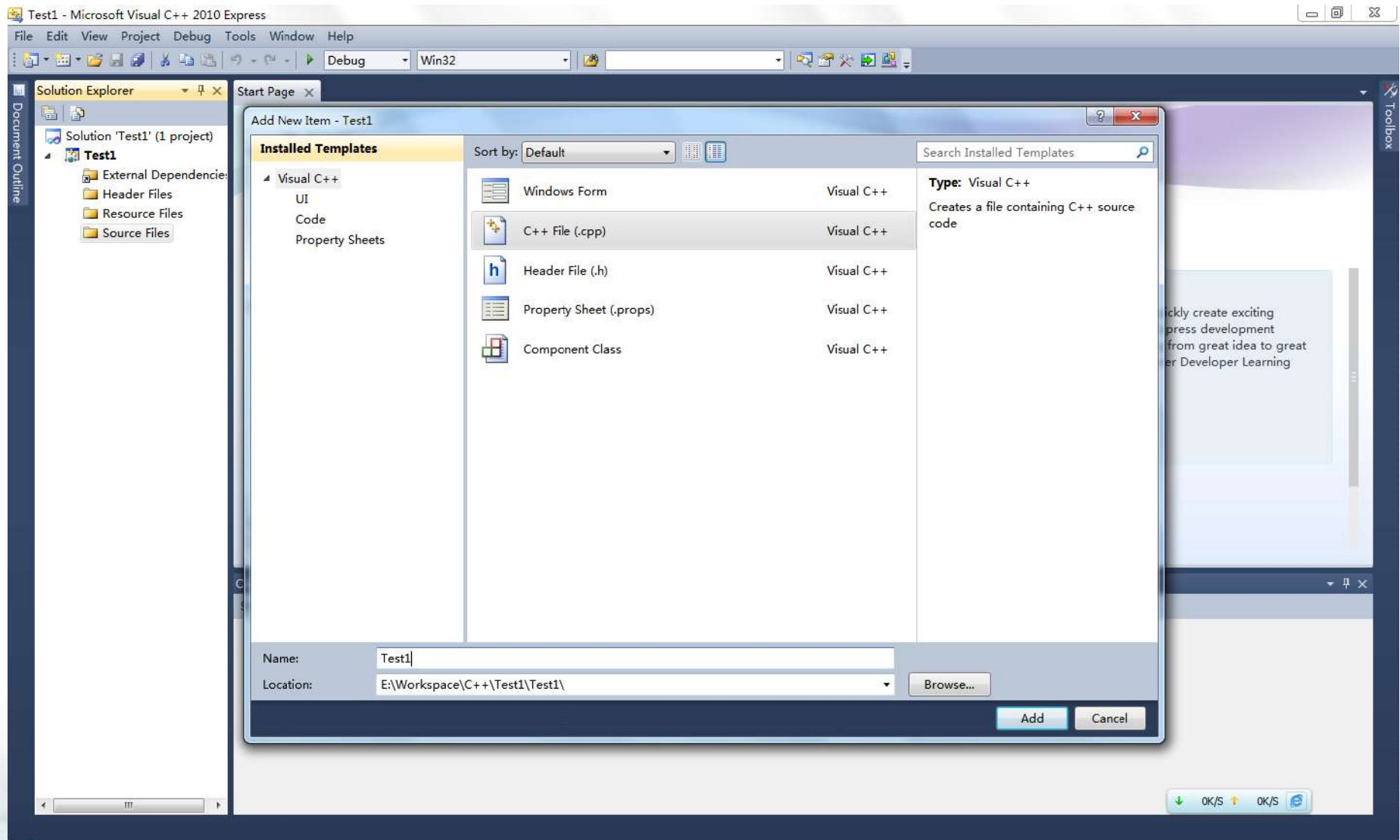




# 创建工程

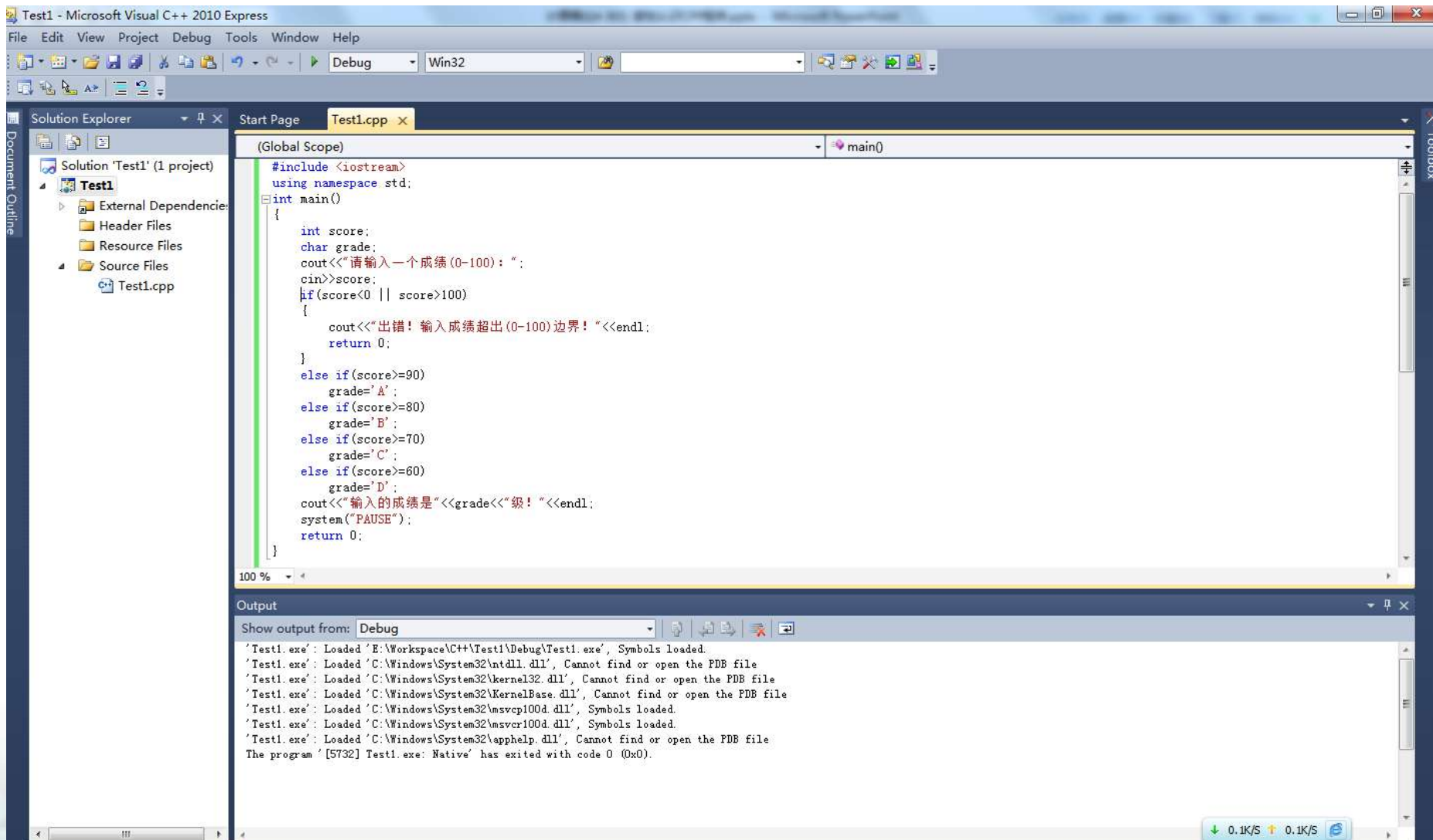


# 创建.cpp文件



清华大学

# Compile & Build



The screenshot displays the Microsoft Visual C++ 2010 Express IDE. The Solution Explorer on the left shows a project named 'Test1' with a single source file 'Test1.cpp'. The main editor window shows the code for 'Test1.cpp', which is a C++ program that prompts the user to enter a score (0-100) and outputs the corresponding grade (A, B, C, D) or an error message if the score is out of range. The Output window at the bottom shows the compilation process, including the loading of various DLLs and the successful execution of the program.

```
#include <iostream>
using namespace std;
int main()
{
    int score;
    char grade;
    cout<<"请输入一个成绩(0-100) : ";
    cin>>score;
    if(score<0 || score>100)
    {
        cout<<"出错! 输入成绩超出(0-100)边界!"<<endl;
        return 0;
    }
    else if(score>=90)
        grade='A';
    else if(score>=80)
        grade='B';
    else if(score>=70)
        grade='C';
    else if(score>=60)
        grade='D';
    cout<<"输入的成绩是"<<grade<<"级!"<<endl;
    system("PAUSE");
    return 0;
}
```

Output:

```
'Test1.exe': Loaded 'E:\Workspace\C++\Test1\Debug\Test1.exe', Symbols loaded.
'Test1.exe': Loaded 'C:\Windows\System32\ntdll.dll', Cannot find or open the PDB file
'Test1.exe': Loaded 'C:\Windows\System32\kernel32.dll', Cannot find or open the PDB file
'Test1.exe': Loaded 'C:\Windows\System32\KernelBase.dll', Cannot find or open the PDB file
'Test1.exe': Loaded 'C:\Windows\System32\msvcrt100d.dll', Symbols loaded.
'Test1.exe': Loaded 'C:\Windows\System32\msvcr100d.dll', Symbols loaded.
'Test1.exe': Loaded 'C:\Windows\System32\apphelp.dll', Cannot find or open the PDB file
The program '[5732] Test1.exe: Native' has exited with code 0 (0x0).
```

# Debug

Test1 (Debugging) - Microsoft Visual C++ 2010 Express

File Edit View Project Debug Tools Window Help

Process: [7188] Test1.exe Thread: [2604] Main Thread Stack Frame: Test1.exe!main() Line 14

Solution Explorer

- Solution 'Test1' (1 project)
- Test1
  - External Dependencies
  - Header Files
  - Resource Files
  - Source Files
    - Test1.cpp

Test1.cpp

```
(Global Scope)
- main()

#include <iostream>
using namespace std;

int main()
{
    int score;
    char grade;
    cout<<"请输入一个成绩(0-100): ";
    cin>>score;
    if(score<0 || score>100)
    {
        cout<<"出错! 输入成绩超出(0-100)边界!"<<endl;
        return 0;
    }
    else if(score>=90)
        grade='A';
    else if(score>=80)
        grade='B';
    else if(score>=70)
        grade='C';
    else if(score>=60)
        grade='D';
    cout<<"输入的成绩是"<<grade<<"级!"<<endl;
    system("PAUSE");
}
```

Watch 1

Name	Value	Type
score	89	int
grade	'-52'?	char

Call Stack

Name	Language
Test1.exe!main() Line 14	C++
Test1.exe!_tmainCRTStartup() Line 555 + 0x19 bytes	C
Test1.exe!mainCRTStartup() Line 371	C
kernel32.dll!758ded6c()	
[Frames below may be incorrect and/or missing, no symbols loaded for kernel32.dll]	
ntdll.dll!7755377b()	
ntdll.dll!7755374e()	

Ready

Autos Locals Threads Modules Watch 1

Call Stack Breakpoints Output

0.1K/s 0K/s



清华大学



# C++程序的加工和执行



- 源程序，不能直接执行，必须先把它转换为可执行程序。

- 加工通常分两步：

- ◆ 编译

- 编译程序处理源程序，生成机器语言目标文件。目标文件不能执行，缺少必要的C++程序运行系统和库功能。

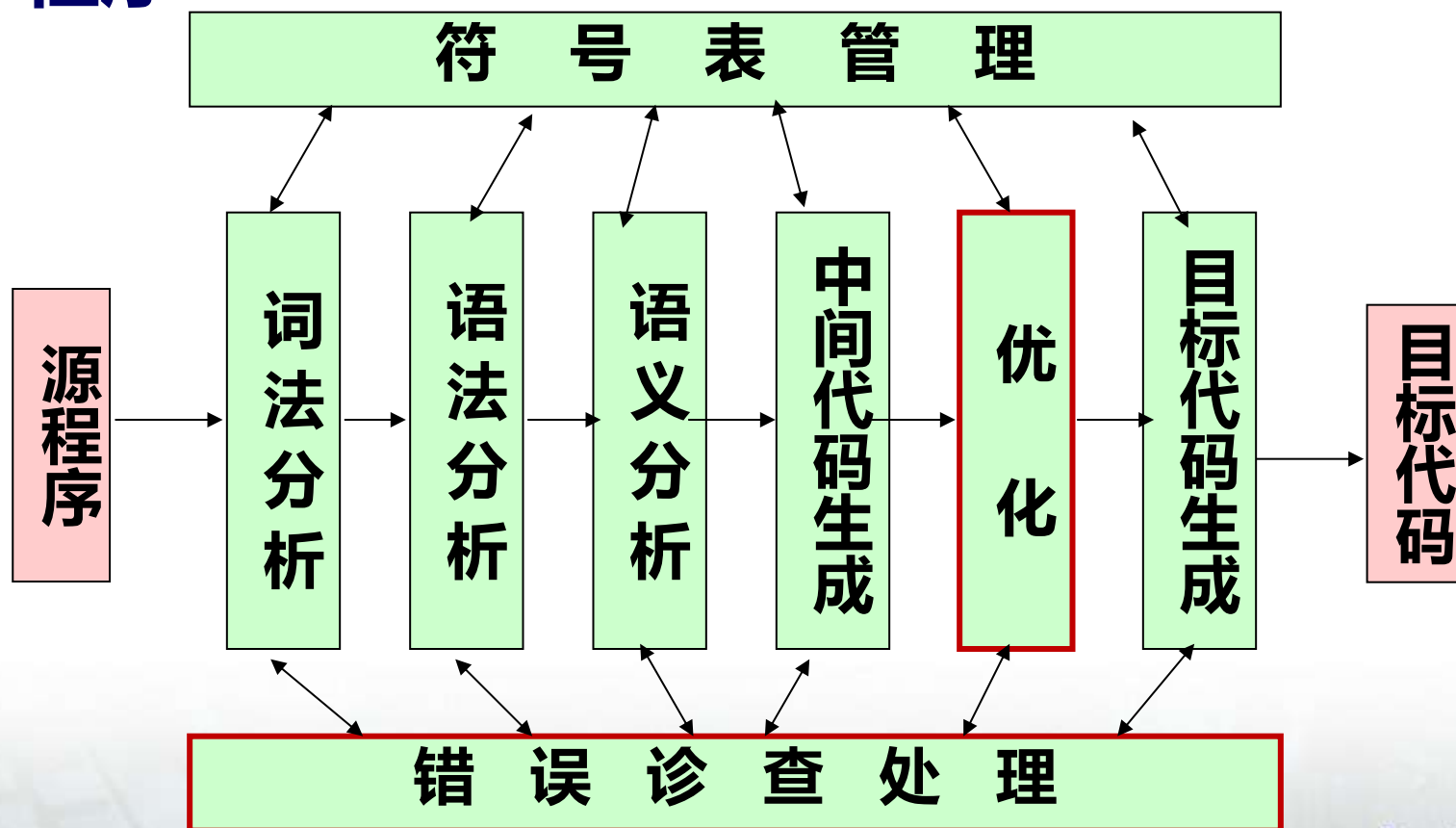
- ◆ 连接

- 把目标模块与运行系统、库模块组合起来，构成完整的可执行程序。



# 编译 (Compilation) 过程

- 把高级语言程序翻译成与其等价的目标语言程序





# C++程序的逻辑结构

## ■ C++程序本身

- ◆ 每个C++程序都由很多个“函数”组成
  - 每个程序都以main()函数开始（程序入口）
  - 每个函数的形式都与int main()函数相似

int	Max	(int x, int y)
↑	↑	↑
函数类型	函数名	参数类型 参数名

- ◆ 如 double pow(double x, double y)
- ◆ 语句分号结尾，一行可以写多个语句，一个语句可以写多行；格式变化不会影响程序运行；
- ◆ 程序中的注释可以放在/\*....\*/之间，或//之后；



北京大学



# 学会利用函数

■ 计算  $\sqrt{2^3 + 4^2}$

■ 用到的数学函数 `sqrt ( x)pow(x,n);`

```
#include <iostream>
```

```
#include <cmath>
```

```
using namespace std;
```

```
int main()
```

```
{
```

```
    int a = 2, b = 3, c = 4, d = 2;
```

```
    int nPow1 = pow(a, b), nPow2 = pow(4, 2);
```

```
    float fResult = sqrt(nPow1+nPow2);
```

```
    cout << fResult;
```

```
    return 0;
```

```
}
```

■ 输出 : 4.89898



北京大学



## 自定义函数

```
#include <iostream>
using namespace std;
int add(int x, int y);    //这是函数的声明 ;
int main( )
{
    int a = 30, b = 40, c = 0;
    cout<<"a="<<a<<" ,b="<<b<<endl;
    c = add(a, b);        //这是函数的调用 ;
    cout<<"a + b = "<<c<<endl;
    return 0;
}
int add(int x, int y)    //这是一个函数的定义 ;
{
    int z=x;
    z=x+y;
    return z;
}
```





# 关于C++程序的书写

## ■ 注意事项：

- ◆ 每行程序的书写必须按照逻辑进行**缩进**；
- ◆ 编写如下程序行时，应该提供**注释**：
  - 变量的定义；//说明变量代表的意义或作用；
  - 函数的定义；//说明该函数的主要作用；
  - 复杂逻辑出现时；//解释一段语句的作用；
- ◆ **变量的命名**应该尽可能体现变量的含义；



```
#include <iostream.h>
void main()
{
    int score;
    char grade;
    cout<<"请输入一个成绩(0-100)";
    cin>>score;
    if(score<0 || score>100)
    {
        cout<<"出错！输入成绩超出(0-100)边界！"<<endl;
        return ;
    }
    else if(score>=90)
        grade='A';
    else if(score>=80)
        grade='B';
    else if(score>=70)
        grade='C';
    else if(score>=60)
        grade='D';
    cout<<"输入的成绩是"<<grade<<"级！"<<endl;
}
```

```
#include <iostream.h>
```

```
void main()
```

```
{
```

```
    int score;
```

```
    char grade;
```

```
    cout<<"请输入一个成绩(0-100)";
```

```
    cin>>score;
```

```
    if(score<0 || score>100)
```

```
    {
```

```
        cout<<"出错！输入成绩超出(0-100)边界！"<<endl;
```

```
        return ;
```

```
    }
```

```
    else if(score>=90)
```

```
        grade='A';
```

```
    else if(score>=80)
```

```
        grade='B';
```

```
    else if(score>=70)
```

```
        grade='C';
```

```
    else if(score>=60)
```

```
        grade='D';
```

```
    cout<<"输入的成绩是"<<grade<<"级！"<<endl;
```

```
}
```

应使用

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

不能出现中文字符（注释除外）

int main() 要求：  
一定要返回一个数字！

int main() 要求：以return 0;结尾

要看到运行结果，可以在return 0;  
之前增加 system( "PAUSE" );

```
#include <iostream>
using namespace std;
int main()
{
    int score;
    char grade;
    cout<<"请输入一个成绩(0-100) : ";
    cin>>score;
    if(score<0 || score>100)
    {
        cout<<"出错！输入成绩超出(0-100)边界！"<<endl;
        return 0;
    }
    else if(score>=90)
        grade='A';
    else if(score>=80)
        grade='B';
    else if(score>=70)
        grade='C';
    else if(score>=60)
        grade='D';
    cout<<"输入的成绩是"<<grade<<"级！"<<endl;
    return 0;
}
```

# 什么样的程序是“好程序”

## 我们重视：

- ✓ “我的程序运行结果正确，解决了问题！”
- ✓ “我的程序更容易被别人看懂！”
- ✓ “我的程序结构更清楚，更美！”

## 我们不重视：

- ✗ “我少使用了几个变量！”
- ✗ “我的程序行数比较少！”
- ✗ “我的程序运行起来快了一些！”

# 怎么学？

从“简单的”开始

从“抄”程序开始

从“模仿”开始

抓大放小，学习狗熊

练习！练习！练习！

程序设计 = K + S



不积跬步  
无以至千里  
不积小流  
无以成江海

荀子（约公元前313 - 前238）

——《劝学》





好好想想，有没有问题？

谢谢！



清华大学