头文件

#include <iostream>

Std命名空间

using namespace std;

注释

// 注释

/\* 注释 \*/

主函数

int main()

{

cout << "Hello World";

return 0;

}

显示内容

cout << "Hello World"; // 打印Hello World之后不换行

cout << "Hello World" << endl; // 打印之后换行

每个内容之前都要有一个 <<

cout << "Hello " << “World！”;

cout << "Hello " << “World！” << "\n";

数据类型

布尔值 bool

字符串 char

整数 int

浮点数 float

双浮点数 double

无类型 void

宽字符 wchar\_t

类型定义 typedef

tpyedef int zhengshu

进而定义整数a就相当于

int a 相当于 zhengshu a

枚举类型：enum

enum 枚举名 {标识符1,标识符2...} 枚举变量

enum color {red, green, blue} c;

c = red;

其中 标识符分别对应 0, 1, ... 如果 cout << c; 则得到

0

如果

enum color {red, green = 5, blue} c;

则 他们分别对应0, 5, 6

定义变量

int a, b, c;

char d, e;

float f, g;

double h;

也可以同时赋值：

int a, b = 1, c;

例子：

#include <iostream>

using namespace std;

int main ()

{

// 变量定义

int a, b;

int c;

float f;

enum color { red, blue = 5, green } g;

g = blue;

// 实际初始化

a = 10;

b = 20;

c = a + b;

cout << c << endl;

f = 70.0/3.0;

cout << f << endl;

cout << g << endl;

return 0;

}

返回的结果

30

23.3333

5

定义函数

先声明函数，然后可以在任意位置定义函数

int func(); // 声明存在函数func

int main()

{

int i = func(); // 函数调用func

}

int func() // 函数定义

{

内容

return 结果;

}

局部变量、全局变量

局部变量：在函数内部声明，只在函数内部有效

全局变量：在函数外部声明

#include <iostream>

using namespace std;

int g; // 全局变量声明

int main ()

{

int a, b; // 局部变量声明

a = 10;

b = 20;

g = a + b;

cout << g;

return 0;

}

如果局部变量和全局变量冲突，则在此局部中，以局部变量为主

#include<iostream>

using namespace std;

int g = 20;

int func();

int main(int argc, char const \*argv[]) {

int k = func(); // 在func里k = g = 10

cout << g << endl; // 出来之后g = 20

cout << k << endl;

return 0;

}

int func(){

int g = 10;

return g;

}

常量

数字：

1, 121, 1.5, 2E-3

布尔值

true, false

不应该把true和false看作1和0,但是在cout打印时，是1和0

字符

|  |  |
| --- | --- |
| 转义序列 | 含义 |
| \\ | \ |
| \' | ' |
| \" | " |
| \? | ? |
| \a | 报警铃声 |
| \b | 退格键 |
| \f | 换页符 |
| \n | 换行符 |
| \r | 回车 |
| \t | 水平制表符 |
| \v | 垂直制表符 |

定义常量

#define 常量名 常量值

例如

#define pi 3.14;

#define c "HALLO";

但是常量不能用 cout打印，例如 cout << pi << endl;

只能赋值后，打印变量

double p = pi;

char q[] = c;

cout << p << endl;

cout << q << endl;