|  |  |  |
| --- | --- | --- |
| 类型 | 含义 | 最小尺寸 |
| Bool | 布尔类型 | 未定义 |
| Char | 字符 | 8位 |
| Wchar\_t | 宽字符 | 16位 |
| Char16\_t | Unicode字符 | 16位 |
| Char32\_t | Unicode字符 | 32位 |
| Short | 短整型 | 16位 |
| Int | 整形 | 16位 |
| Long | 长整形 | 32位 |
| Long | 长整形 | 64位 |
| Float | 单精度浮点数 | 6位有效数字 |
| Double | 双精度浮点数 | 10位有效数字 |
| Long double | 扩展精度浮点数 | 10位有效数字 |

new 为指针分配内存

typeName \* pointer\_name = new typeName [typeCount] （初始化数量）

delete 释放指针上的内存

delete ps;

成员函数at()

Arrayname.at(下标)

iostream

**cin:**输入对象 标准输入 在获取字符数组输入时只读取一个单词 会将换行符保留在缓存区中

**cout:**输出对象 标准输出

**cerr:**输出警告和错误消息 标准错误

**clog:**输出程序运行时的一般性信息

**cin.put():**成员函数提供了另一种显示字符串的方法，可以代替<<运算符

**cin.getline():**类 读取一行输入，直到到达换行符，然后丢弃换行符。

**cin.getline()**接受2个参数，第一个是存储输入行的数组，第二个是要读取的字符数

**cin.get():**函数读取一行输入，直到到达换行符，然后将换行符保留在缓存区中。接受2个参数同getline。还可以接受一个参数char类型同getchar()

变种 cin.get(name,ArSize).get();

(cin>>name).get();

getline(cin,stringname);

cin.eof()如果检测到了EOF返回true，否则返回false

cin.fail()同理 在文件中，最后一次读取操作中发生了类型不匹配的情况，将返回true

cin.clear()清空错误的输入流

open()方法 打开文件

close()方法 关闭文件

写入文件声明方法类型

ofstream

读取文件声明方法类型

ifstream

文件函数

文件对象.open(文件名字)

is\_open()方法 检测文件是否被正确打开

bad()方法 如果最后一次读取文件时发生了意外问题，返回true

good()方法 指出最后一次读取输入的操作是否成功

auto自动类型推断功能

可以用他开自动初始化函数指针

cout.precision()输出流格式控制函数，确定浮点数精度值显示

showpos 正数前面加上+号

fixed 使用小数计数法

scientific 使用科学计数法

uppercase 使用大写字符

showbase 显示数字的进制

boolalpha bool值使用字符表示 , true或者false

noboolalpha bool使用0和1表示

left 靠左对齐

right 靠右对齐

internal 字符靠左对齐, 数字卡右对齐

cout.setf()通过设置格式标志来控制输出形式 参数在ios\_base类下

boolalpha 可以使用单词”true”和”false”进行输入/输出的布尔值.

oct 用八进制格式显示数值.

dec 用十进制格式显示数值.

hex 用十六进制格式显示数值.

left 输出调整为左对齐.

right 输出调整为右对齐.

scientific 用科学记数法显示浮点数.

fixed 用正常的记数方法显示浮点数(与科学计数法相对应).

showbase 输出时显示所有数值的基数.

showpoint 显示小数点和额外的零，即使不需要.

showpos 在非负数值前面显示”＋（正号）”.

skipws 当从一个流进行读取时，跳过空白字符(spaces, tabs, newlines).

unitbuf 在每次插入以后，清空缓冲区.

internal 将填充字符回到符号和数值之间.

uppercase 以大写的形式显示科学记数法中的”e”和十六进制格式的”x”.

decltype()关键字提供模板函数形参的数据类型，括号内为数据类型，如果括号内是一个函数，并不会调用函数

重载运算符

Operator op(argument-list)

Op必须是c++的运算符

例如 operator+()重载+运算符 或者 operator\*()重载\*运算符

cstdlib

NULL预处理变量

exit()函数 接受参数EXIT\_FAILURE终止程序

climits

cstring同string.h

**strcat():**将字符串附加到字符数组末尾

**strncat():**接受第三个参数，指定了字符串的长度

**strcpy():**将字符串复制到字符数组中

**strncpy():**接受第三个参数，指定了字符串的长度

**strlen():**计算字符串的长度

**ArrerName.size():**同strlen()

string

string类

vector

vector对象

创建一个名为vt的vector对象，他可以存储n\_elem个类型为typeName的元素

vector<typeName> vt(n\_elem)

参数n\_elem可以是整形常量，也可以是变量

array

声明创建一个名为arr的array对象，他包含n\_elem个类型为typename的元素

array<typeName,n\_elem>arr;

n\_elem不能是变量

ctime同time.h

clock()返回程序开始执行后所用的系统时间，返回类型是clock\_t

CLOCKS\_PER\_SEC 该常量等于每秒钟包含的系统时间单位数

Cctype同ctype.h

isalpha(ch)如果ch是一个字母，函数返回非零，否则返回0

ispunct(ch)如果ch是标点，函数返回true，否则返回0

isalnum() 参数是字母或者数字 返回true

iscntrl() 参数是控制字符，返回true

isdigit() 参数是数字 返回true

isgraph() 参数是除了空格之外的打印字符，返回true

islower() 参数是小写字母

isprint() 参数是打印字符（包括空格）

isspace() 参数是标准空白字符，如空格、进制、换行符、回车、水平制表符或垂直制表符

isupper() 参数是大写字母

isxdigit() 参数是十六进制数字

tolower() 将大写转换小写

toupper() 将小写转换大写

iomanip

setw()函数 是c++中在输出操作中使用的字段宽度设置，设置输出的域宽，n表示字段宽度。只对紧接着的输出有效，紧接着的输出结束后又变回默认的域宽。当后面紧跟着的输出字段长度小于n的时候，在该字段前面用空格补齐；当输出字段长度大于n时，全部整体输出。

函数指针

类型名 (\*函数名)(参数类型名);

内联函数

Inline

引用变量

Int & rodents = rats rodents作为rats变量的别名

内存模型

auto 自动类型判断

register 请求寄存器存储

static 作用域为整个文件

extern 引用声明，即声明引用在其他地方定义的变量

thread\_local（c++） 支出变量的持续性与其所属线程的持续性相同

mutable 她可以用来指出及时结构变量为const但是某个成员可以被修改

volatile 即是程序代码没有对内存单元进行修改，其值也可能发生变化。防止编译器让变量的值存到寄存器中，阻止编译器对齐进行优化

编程规范

声明：

类：首字母大写+前置下划线

结构体：首字母大写+后置下划线

宏：所有字母大写

const：所有字母大写

变量名：所有字母小写

私有成员：第二个单词字母大写

stack