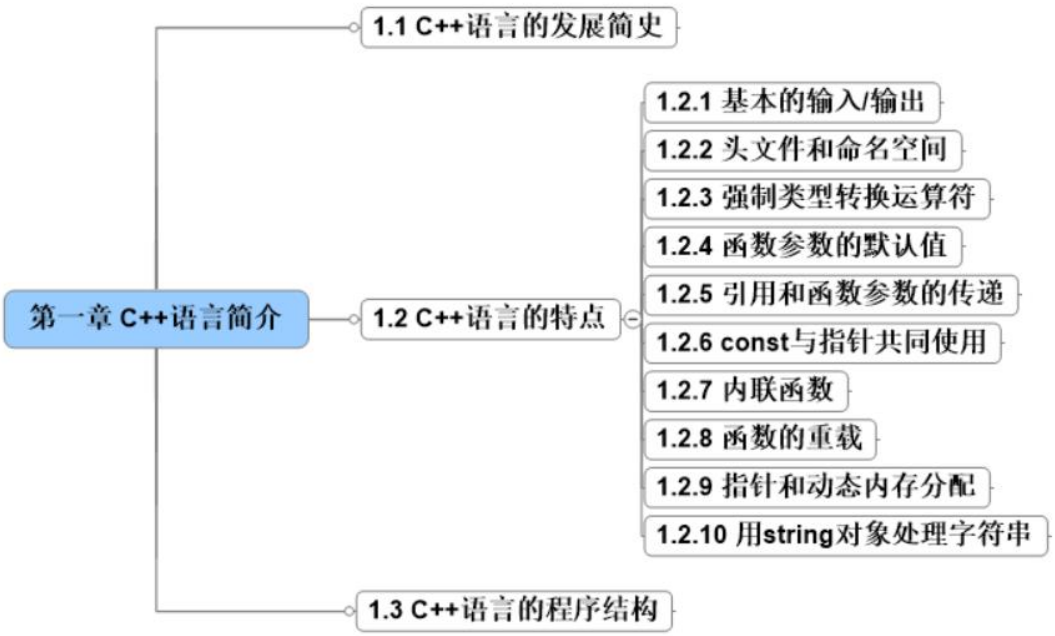
C++程序设计第二节课官方笔记

目录

一、课件下载及重播方法

二、教材节构图



三、本章知识点及考频总结

（一）选择题（共 8 道）

1.引用相当于给变量起了一个别名。变量对应于某个内存地址，如果给某个变量起了别名，相当于变量和这个引用都对应到同一地址。程序中使用哪个名字都是允许的。在 C++中，“引用”的定义格式如下：

```
类型名 &引用名=同类型的某变量名;
```

举例如下：

```
int oneInt;
int &aname=oneInt;  //声明引用
```

定义引用时，可以在定义的前面加 const 关键字，表明该引用是“常引用”。例如：

```
int oneInt;
const int &cname=oneInt;
```

2.引用在函数中的使用

在程序中不仅能定义变量的引用，还可以将引用用在函数中。引用既可以作为函数的参数使用，也可以作为函数的返回值使用。

在 C++中，函数调用时参数的传递有两种方式：**传值**和**传引用**。

传值，实际上是传递对象的值。传引用是传递对象的**首地址值**。如果函数的形参不是引用，那么调用时实参传递给形参通常采用的是传值的方式，即将实参的值拷贝给形参。在函数执行过程中，都是对这个拷贝进行操作的，函数执行完毕返回后，形参的值并不拷贝回实参，也就是说函数内部对形参的改变不会影响到函数外实参的值。

如果函数的形参是引用，则调用时实参传递给形参采用的是**传引用**的方式。函数调用时，实参对象名传递给形参对象名，形参对象名就成为实参对象名的别名，即形参是对应实参的引用，它们是等价的，代表同一个对象，也可以看作是将实参的地址传递给了形参。在函数内部对形参的操作，都是对这个地址的内容进行的，相当于对实参的值进行了操作。所以当函数执行完毕返回后，实参的变化被保留下来。

在**传值调用函数的情况下，形参是实参的副本，形参的改变不会影响到实参**。但调用时生成临时变量完成值的复制，这需要一定的时间和空间。**在传引用调用函数的情况下，形参的改变就意味着实参的改变。**

3. 设有说明 const char * const p="ABCD";，禁止修改指针 p 本身，且禁止通过 p 修改所指向的数据。

4.为了避免频繁的函数调用与返回，C++语言引入了内联函数的概念。使用内联函数，编译器在编译时并不生成函数调用，而是将程序中出现的每一个内联函数的调用表达式直接用该内联函数的函数体进行替换，就像整个函数体在调用处被重写了一遍一样。很显然，使用内联函数会使最终可执行程序的体积增大。这是以空间消耗节省时间开销。

内联函数应该定义在前，调用在后，定义时只需在函数头返回值类型的前面加上关键字 inline。

内联函数主要应用于代码量少的函数，编译时，编译程序将整个函数体的代码复制到调用该函数的位置，而不会编译成函数调用的指令。也就是说，内联函数不是在调用时发生控制转换，而是在编译时将函数体嵌入在每一个调用处。

如果函数体中有循环语句和 switch 语句则通常不定义为内联函数。

5.所谓函数重载，是指在程序的同一范围内声明几个功能类似的同名函数。

实现函数的重载必须满足下列条件之一：

- 参数表中对应的参数类型不同。
- 参数表中参数个数不同。

如果函数参数表中不同类型参数的次序不同，也符合上面所说的条件。要注意的是，返回值类型不能用来区分函数，也就是说，如果两个函数的名字和参数表都是一样的，仅仅是返回值类型不同，则这两个函数不是重载的，编译器认为它们是重复定义，编译时会报错。

另外，采用引用参数也不能区分函数，见例 1-9。

例 1-9 错误的重载函数

```
void print(double);
```

```
void print(double&);          //错误!
```

6.指针变量中保存的是一个地址，有时也称指针指向一个地址。

数组的长度是声明数组时指定的，在整个程序运行过程中通常是不变化的。C++语言不允许定义元素个数不确定的数组。例如：

```
int n;
int a[n];          //在有些编译环境下，这种定义是不允许的，因为 n 未知
```

在 C++语言中，使用 new 运算符实现动态内存分配。例如，可以写如下的语句：

```
p=new T;
```

其中，T 是任意类型名，p 是类型为 T*的指针。这样的语句会动态分配出一片大小为 sizeof(T)字节的内存空间，并且将该内存空间的起始地址赋值给指针 p。例如：

```
int * p;
p=new int;
*p=5;
```

第 2 行语句动态分配了有 4 个字节大小的内存空间，p 指向这个空间的首地址，之后通过指针 p 可以读写该内存空间。第 3 行语句是向这个空间中存入数值 5。

使用 new 运算符还可以动态分配一个任意大小的数组：

```
p=new T[N];
```

其中，T 是任意类型名，p 是类型为 T*的指针，N 代表数组“元素个数”，可以是任何的值为正整数的表达式。数组中元素的类型是 T 类型。这条语句动态分配了 N*sizeof(T)个字节的内存空间，指针 p 指向这段空间的首地址。

使用 new 运算符动态申请的内存空间，需要在使用完毕释放。C++提供了 delete 运算符，用来释放动态分配的内存空间。delete 运算符的基本用法如下：

```
delete 指针;
```

如果是使用 new 运算符动态分配了一个数组，那么释放该数组时，语句如下：

```
delete []指针;
```

例如，释放例 1-14 中分配的数组空间的语句是“delete [] pArray;”。

使用 new 运算符动态分配的内存空间，一定要用 delete 运算符释放。

7.声明 string 对象

```
string str1;          //声明 string 对象 str1，值为空
string city="Beijing";//声明 string 对象 city，并使用字符串常量进行初始化
string str2=city;     //声明 string 对象 str2，并使用字符串变量进行初始化
cout<<"str1="<<str1<<"."<<endl;
cout<<city<<","<<str2<<endl;
```

还可以使用字符数组对 string 变量进行初始化。例如:

```
char name[]="C++程序";
string s1=name;
```

还可以声明一个 string 对象数组，即数组中每个元素都是字符串。例如：

```
string citys[]={"Beijing","Shanghai","Tianjin","Chongqing"};
cout<<citys[1]<<endl;                  //输出 Shanghai，数组下标从 0 开始
cout<<sizeof(citys)/sizeof(string)<<endl;       //输出数组元素个数
```

最后一行语句将输出数组元素个数。citys 是 string 对象数组，sizeof(citys)是整个数组占用的空间大小，sizeof(string)是每个 string 对象的大小，所以 sizeof(citys)/sizeof(string)表示的是数组元素个数。

8.C++程序以.cpp 作为文件扩展名，文件中包含若干个类和若干个函数。程序中必须有且仅

有一个主函数 main( )，这是程序执行的总入口。主函数也称为主程序。程序从主函数的开始处执行，按照其控制结构，一直执行到结束。

程序的**结束**通常是遇到了以下两种情形之一。

1)在主函数中遇到 return 语句。

2)执行到主函数最后面的括号}。

主函数中可以调用程序中定义的其他函数，但其他函数不能调用主函数。主函数仅是系统为执行程序时所调用的。

C++程序中，仍沿用 C 语言的注释风格，即**注释**有以下两种形式。

1)从/*开始，到*/结束，这之间的所有内容都视作注释。

2)从//直到行尾，都是注释。

(二) 主观题 (共 1 道)

```
#include<iostream>
#include<string>
using namespace std;
int main()
{   string str;     //未初始化，空串
    if(str.empty())
        cout<<"str is NULL. "<<",length="<<str.length()<<endl;
    else
        cout<<"str is not NULL. "<<endl;
    str=str.append(“abcdefg”);
    cout<<"str is"<<str<<", size="<<str.size()<<endl;
    const char *p=str.c_str();
    cout<<"p="<<p<<endl;
    cout<<"find:"<<str.find("de",0)<<endl;      //查找成功，3
    cout<<"find:"<<str.find("de",4)<<endl;      //查找失败
    string str1=str.insert(4, "123");
    cout<<str1<<endl;
    return 0;
}
```

程序的执行结果如下:

```
str is NULL.,length=0
str is abcdefg,size=7
p=abcdefg
find:3
find:18446744073709551615
abcd123efg
```

四、配套练习题

1、对指针动态分配空间用的关键字是（　　）。

A:define

B:int

C:new

D:float

2、若有定义 int *p=new int（0），则下列说法正确的是（　　）。

A:系统用指针变量 p 来表示所指整型变量

B:声明一个指针变量 p，指向名为 new 的存储单元

C:系统为指针变量 p 分配一个整型数据的存储空间

D:通过运算符 new，分配一个整型数据的存储空间，并将其内存地址赋予指针变量

3、若有说明：int n=2,*p=&n,*q=p;，则以下非法的赋值语句是（　　）。

A:n=*q

B:p=n

C:p=q

D:*q=*p

[参考答案] CDB

五、其余课程安排