

## 一

当用（`#define`）宏定义的指针变量类型在一条语句中同时创建多个变量时，宏定义的指针变量类型只对第一个变量有用

用`typedef`重命名变量类型名（关键字），再在一条语句中同时创建多个变量时，定义的指针变量类型对多个变量有用

## 二

未初始化的全局变量和局部变量，（全局变量）会被系统默认初始化为0

## 三

宏定义的宏实在预处理阶段完成的

## 四

C语言函数值类型缺省定义

C中：如果函数未指定返回值类型，则默认为int

c++中：如果一个函数没有返回值，返回值类型必须指定为void

## 五

有A、B、C、D四个人，要在夜里过一座桥。他们通过这座桥分别需要耗时1、2、5、10分钟，只有一支手电，并且同时最多只能两个人一起过桥。请问，如何安排，能够在17分钟内这四个人都过桥？

第一趟过去：两人：花1分钟和2分钟的人 //总共耗时：2分钟

第一趟过来：一人：花1分钟的人 //总共耗时：3分钟

第二趟过去：两人：花5分钟和10分钟的人 //总共耗时：13分钟

第二趟过来：一人：花2分钟的人 //总共耗时：15分钟

第三趟过去：两人：花1分钟和2分钟的人 //总共耗时：17分钟

## 六

### 函数设计应该追求高内聚低耦合

【高内聚、低耦合】

内聚：

每个模块尽可能独立完成自己的功能，不依赖于模块外部的代码。

耦合：

模块与模块之间接口的复杂程度，模块之间联系越复杂耦合度越高，牵一发而动全身。

目的：

使得模块的“可重用性”、“移植性”大大增强 通常程序结构中各模块的内聚程度越高，模块间的耦合程度就越低

模块粒度：

『函数』 高内聚：尽可能类的每个成员方法只完成一件事（最大限度的聚合） 低耦合：减少类内部，一个成员方法调用另一个成员方法 『类』 高内聚低耦合：减少类内部，对其他类的调用

函数设计应做到低耦合，高内聚。也就是说，在不增加代码复杂度的情况下，尽量减少函数之间的调用关系，在本函数实现规定的功能。

“低耦合，高内聚”的函数设计有什么好处呢？好处有以下几点：

第一，便于对程序进行维护。这点很重要，特别是刚入职的员工，如果他们阅读到了逻辑清晰、编程规范的代码，真的是一种福气。

第二，便于程序版本的演进。有了好的“榜样”，之后对程序的增删改的工作都更加的容易了。

第三，便于不同项目组或产品线之间的沟通交流。优秀的代码应该拿出来，供大家一起学习。“他山之石，可以攻玉”，只有不断地学习别人好的、成功的经验，自己的能力才能够得到提升。

## 七

### 野指针的产生原因

情况一

原因

指针变量声明时没有被初始化。

解决办法

指针声明时初始化，可以是具体的地址值，也可让它指向NULL。

情况二

原因

指针 p 被 free 或者 delete 之后，没有置为 NULL。

解决办法

指针指向的内存空间被释放后指针应该指向NULL。

情况三

原因

指针操作超越了变量的作用范围。

## 八

重载插入运算符<<时，其运算符函数的返回值类型应当是\_\_\_\_\_

答案

重载插入运算符<<时，其运算符函数的返回值类型应当是ostream

重载抽取运算符>>时，其运算符函数的返回值类型应当是istream

## 九

for循环只能用于循环次数已经确定的情况、

error:

```
char str[100];
```

```
gets(str);
```

```
for (int i=0;i<strlen(str);i++)
```

```
{  
printf("%c ",str[i]);  
}
```

在没有输入字符串之前 并不知道要循环多少次

十

十一

十二

十三

十四

十五