百度：

1. 自我介绍
2. 一个类，如何禁止构造（禁止调用其构造函数）

解答：将构造函数声明为私有成员函数

1. 打印64位int

不同的编译打印方式不同VC printf(“%l64d”,a);

G++ printf(“%lld”,a);

1. Std:: vector的底层数据结构（其他的呢，map,list）

动态数组，假设刚开始vector分配的大小为size,当要添加第size+1个元素时，vector大小增大为2\*size. 当动态数组内的元素比动态数组长度多一时，动态数组长度翻倍！

1. 一个数组，只知道其数组名，不知道类型，求其长度

Sizeof(a)/sizeof(a[0]);

1. New和malloc 的区别
2. Malloc/Free 是C/C++的标准库函数，new/delete是C++的运算符，因此编译器无法控制库函数。
3. Malloc/Free不会调用构造/析构函数，New/delete会调用(C++是弱类型，动态类型)
4. New出来的是带有类型的指针，malloc返回的是void指针
5. New操作符会从自由存储区为对象动态分配内存，malloc是在堆上自由分配内存。
6. C++内存泄漏
7. new/malloc分配了内存空间，但是并没有delete或free
8. 其他的呢
9. Linux命令

ls，cd.grep,cp,mv,rm,kill,tar,cat.chmod,vim,gcc

1. HTTP请求

200 请求成功

301 资源转移到了其他URL

401 身份认证

402 保留

403 拒绝请求

404 请求的资源不存在

500 内部服务器错误

1. 算法题，

**题目**：输入一个整形数组，数组里有正数也有负数。数组中连续的一个或多个整数组成一个子数组，每个子数组都有一个和。求所有子数组的和的最大值。要求时间复杂度为O(n)。

|  |
| --- |
| #include <stdlib.h>  #include <iostream>  #include <vector>  using namespace std;  #define ARR\_LEN(Arr,len){len = sizeof(Arr)/sizeof(Arr[0]);}  int Count(int a[],int n)  {  //sum[i+1] = max(sum[i]+a[i+1],a[i+1])  //根据sum[i],就可以判断sum[i+1]的值  int ans = 0;  int sum[n] = {0};  sum[0]=a[0];  for(int i =1 ;i<n;i++){  sum[i]=a[i]+sum[i-1];  if(sum[i]<0)sum[i]=0; //如果cur大于0意味着，前面一段的子数组和是正反馈，所以需要保留  //如果cur小于0意味着，前面一段的子数组和是负反馈，所以需要重新开始  if(sum[i]>ans)ans =sum[i];  }  return ans;  }  int max(int i,int j){  if(i>j)return i;  return j;  }  int myCount(int a[],int n){ //比较好理解怎么dp的  int ans =0;  int sum[n]={0};  sum[0]=a[0];  for(int i=1;i<n;i++){  sum[i] = max(sum[i-1]+a[i],a[i]);  if(sum[i]>ans) ans = sum[i];  }  return ans;  }说  int main(int argc,char\*\* argv)  {  //int\* a = new int[10]();  int a[] = {1,-2,-3,-5,8,6,-10,5,11};  int len;  ARR\_LEN(a,len);  int ans = Count(a,len);  int ans2 = myCount(a,len);  cout<<ans<<endl;  cout<<ans2<<endl;  //int ans = Count(a,len);  // cout<<ans<<endl;  } |