

C++面试笔试

指针和引用的区别？

- 1、引用必须初始化且只能初始化为另一个变量。指针可不初始化，初始化的时候，可以指向一个地址，也可以为空。
- 2、引用初始化以后不能被改变，指针可以改变所指的对象。
- 3、引用不需要分配内存区域，指针要分配内存区域。

```
/*
 * 输入10个整数，将其中最小的数与第一个数对换，把最大的数与最后一个数对换。
 * 编写3个函数：（1）输入10个数；（2）进行处理；（3）输出10个数。
 */
#include<iostream>
using namespace std;

/*
 * 用符号常量定义数组长度（const:constant）
 *
 * 说明：
 * 由于使用 const 语句定义符号常量带有数据类型，以便系统进行类型检查，
 * 同时该语句具有计算初值表达式和赋初值的功能，所以比宏命令(#define)定义符号常量要优越得多，
 * 因此提倡使用 const 语句。
 */
const int arrayLength=10;

void inputNum(int number[],int temp_length);
void outputNum(int number[],int temp_length);
void swap(int number[],int temp_length);

int main()
{
    int number[arrayLength];
    cout<<"请输入10个整数："<<endl;
    inputNum(number, arrayLength);
    swap(number, arrayLength);
    outputNum(number, arrayLength);
}
```

```

        return 0;
    }

    /*
    *   输入整数
    */
void inputNum(int number[],int temp_length)
{
    int length=0;
    length=temp_length;
    for(int i=0;i<length;i++)
    {
        cin>>number[i];
    }
}

/*
*   交换
*/
void swap(int number[],int temp_length)
{
    int *max,*min,*pointer,temp,length;
    //将数组首地址赋值给指针变量，即指针变量指向数组的第一个元素
    max=min=number;
    temp=0;
    length=temp_length;
    /*
    *   循环变量初始值：数组第2个值的地址；循环条件：数组最后一个值的地址；循环
    变量增值：地址+1
    */
    for(pointer=number+1;pointer<number+length;pointer++)
    {
        if(*pointer>*max)
        {
            //将大数地址赋给max
            max=pointer;
        }
        else if(*pointer<*min)

```

```

        {
            //将小数地址赋给min
            min=pointer;
        }
    }
    //将最小数与第一个数交换
    temp=number[0];
    number[0]=*min;
    *min=temp;
    //将最大数与最后一个数交换
    temp=number[9];
    number[9]=*max;
    *max=temp;
}

```

```

/*

```

```

 * 输出整数

```

```

*/

```

```

void outputNum(int number[],int temp_length)

```

```

{
    int length=0;
    length=temp_length;
    for(int i=0;i<length;i++)
    {
        cout<<number[i]<<"\t";
    }
}

```

```

////////////////////////////////////
////////////////////////////////////

```

实现strlen。

```

int strlen(const char * str) {
    int length = 0;
    while( *str++ ) {
        ++length;
    }
    return length;
}

```

实现strcpy。

```
char* strcpy(char* dst, char* src) {
    char* cp = dst;
    while( *cp++ = *src++ ) {
        // empty
    } // Copy src over dst
    return dst;
}
```

实现strcat。

```
char* strcat(char* dst, char* src) {
    char* cp = dst;
    while( *cp ) {
        ++cp;
    } // Find end of dst
    while( *cp++ = *src++ ) {
        // empty
    } // Copy src to end of dst
    return dst;
}
```

实现strcmp。

```
int strcmp(unsigned char *src, unsigned char *dst) {
    int ret = 0 ;
    while ( ! (ret = *src - *dst) && *dst )
        ++src, ++dst;
    if ( ret < 0 )
        ret = -1 ;
    else if ( ret > 0 )
        ret = 1 ;
    return ret;
}
```

Heap与Stack的区别？

多线程编程最大的问题是什么？如何能写出稳定高效的多线程程序？

什么函数不能声明为虚函数？

构造函数。

1、要构造一个对象，必须清楚地知道要构造什么。

2、防止父类的构造函数被覆盖。

介绍STL。

STL由容器、算法和迭代器组成。STL主要有以下优点：

- 1、可以方便容易地实现搜索数据和排序数据等一系列算法。
- 2、调试程序时更加安全和方便。
- 3、跨平台。

类和结构的区别。

- 1、实例化的类存储在内存的堆内，而结构存储在栈内，结构的执行效率相对较高。
- 2、结构没有析构函数。
- 3、结构不可以继承。

面向对象的特征。

封装、继承、多态。

抽象类及它的用途。

包含纯虚函数的类称为抽象类。抽象类把有共同属性或方法的对象抽象成一个类。

什么是多态？多态的作用？

多态是一种行为的封装，简单地说就是“一个接口，多种实现”，也就是同一种事物所表现出的多种形态。

多态的作用：

- 1、隐藏实现细节，使得代码能够模块化。
- 2、接口可以重用。

在C++中如何实现多态？

- 1、基于继承机制和虚函数来实现。
- 2、使用函数重载。
- 3、使用模板函数。

复制构造函数。

有3中情况会调用复制构造函数：

- 1、一个对象以值传递的方式传入函数体。
- 2、一个对象以值传递的方式从函数返回。
- 3、一个对象需要通过另外一个对象进行初始化。

虚函数和纯虚函数的区别。

纯虚函数是虚函数的一个子集，用于抽象类，含有纯虚函数的类就是抽象类，它不能被实例

化。

如何使用纯虚函数。

纯虚函数用来定义没有实现的方法，用于定义抽象类中需要交给派生类具体实现的方法。

指针形参和引用形参的区别。

指针形参是指函数的参数是指针，它不会像引用形参那样通过函数调用影响实参的值，但是调用后它会修改实参的对象。

静态函数能访问类的私有成员吗？

静态函数只能直接访问类的静态私有成员，静态函数不可以直接访问类的非静态私有成员，但是可以通过自定义的一些特殊方法比如宏替换访问类的非静态私有成员。

一个类可以访问另一个类的私有成员吗？

外部类可以使用宏定义等特殊方法来实现访问类的私有成员。

什么是泛型编程。

所谓的泛型编程，就是以独立于任何特定类的方式编写代码，针对不同的类型提供通用的实现。

C++如何实现泛型编程。

C++中泛型编程的实现是使用C++中的模板技术来实现的，主要是设计函数模板和类模板。

为什么常常把具有层次结构的顶级父类的析构函数定义为虚函数。

为了能正确地调用对象的析构函数。在delete一个抽象类指针时，可以通过虚函数找到真正的析构函数。

重载和重写的区别。

- 1、从定义来讲，重载是指允许存在多个同名函数，而这些函数的参数表不同。重写是指子类重新定义父类虚函数的方法。
- 2、从实现来讲，重载是编译器根据函数不同的参数表，对同名函数的名称做修饰，然后这些同名函数就成了不同的函数。而重写是当子类重新定义了父类的虚函数后，父类指针根据赋给它的不同的子类指针，动态的调用属于子类的该函数。这样的函数调用是在编译期间无法确定的，只能在运行期间绑定。

C++中的空类默认会产生哪些成员函数？

默认构造函数、析构函数、拷贝构造函数、赋值函数、默认取址运算符、默认取址运算符const。

