Java和c++的不同：

Java是一门面向对象的编程语言，不仅吸收了C++语言的各种优点，还摒弃了C++里难以理解的多继承、指针等概念，因此Java语言具有功能强大和简单易用两个特征。Java语言作为静态面向对象编程语言的代表，极好地实现了面向对象理论，允许程序员以优雅的思维方式进行复杂的编程。

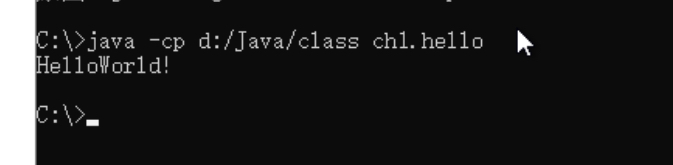
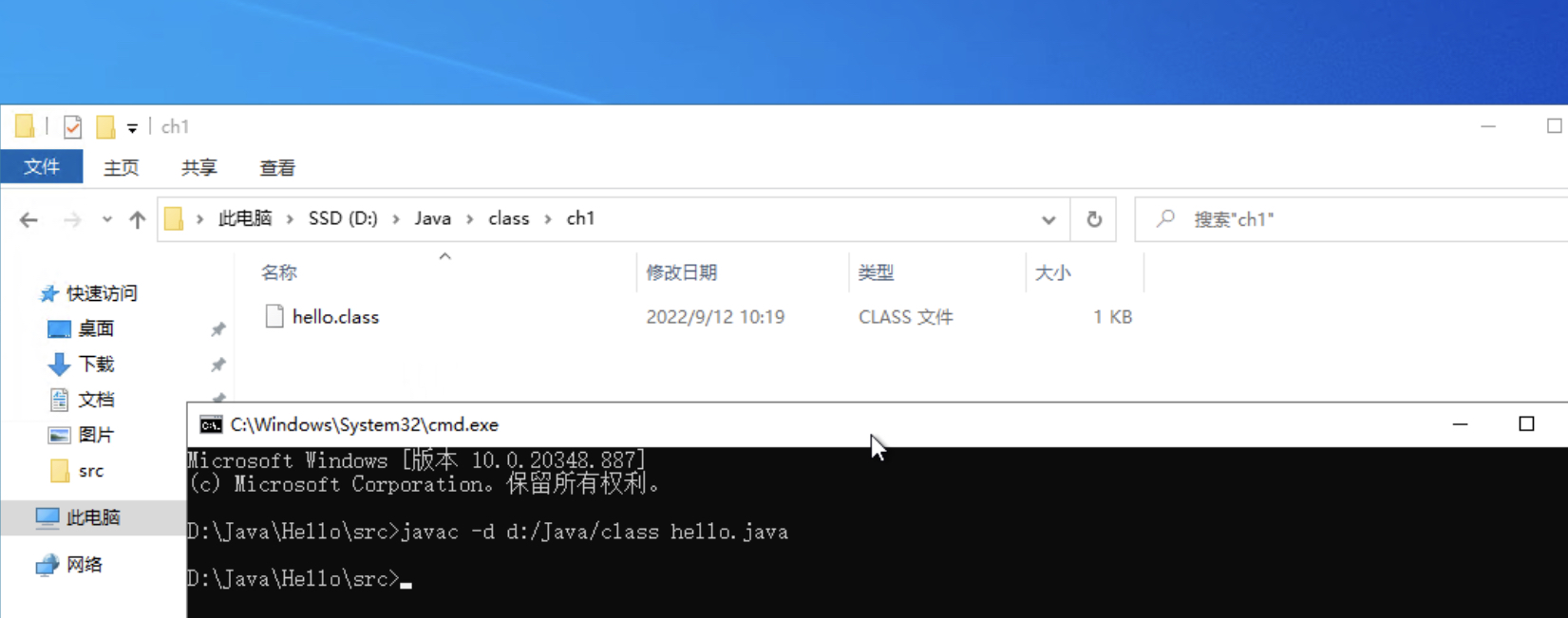
C++既可以进行C语言的过程化程序设计，又可以进行以抽象数据类型为特点的基于对象的程序设计，还可以进行以继承和多态为特点的面向对象的程序设计。C++擅长面向对象程序设计的同时，还可以进行基于过程的程序设计。

世上先有C++，然后才有Java，整个Java语言的发展历史就是一部对C++的填坑史。Java无游离于class之外的零散的东西，没有goto语句，无指针，有垃圾回收机制，数据类型的存储长度有一致性，没有头文件，结构和预处理，新增包的概念。

单文件的编译时间java比C++语言快；C++语言可以直接操作内存，java不能直接操作；C++语言可以封装动态库，java不行；C++语言的代码不容易跨平台，java的代码容易跨平台；C++语言有指针，java没有指针；C++语言可以直接操作串口，java需要第三方jar包支持；C++语言的线程更加灵活，java的线程都已经封装好了；C++可用的标识符为数字、大小写字母、下划线，不能以数字开头；C++语言做单独功能，可以增加效率，java适用做web应用开发。

Java用来操纵对象的引用可以先初始化再指向对象，而C++的引用必须在初始化时就指向对象。Java类除static和final外其他函数定义默认是虚函数，而C++用virtual关键字定义定义函数才是虚函数。Java有一套继承自object类型的单根继承结构，而C++没有标准库内置的类继承体系。Java只能通过new在堆上创建对象，基本类型除外，而C++除了new还是可以创建全局对象或者栈对象。Java有自带垃圾回收机制，不用关心对象的内存释放，而C++需要手动释放通过new创建的对象。Java数组成员能初始化默认值null，提供未初始化的访问和越界访问保护，而C++不提供这个机制。Java局部作用域定义的变量不能隐藏更大作用域定义的变量，而C++可以隐藏。Java类的基本类型成员变量可以自动初始化为默认值，类的复杂类型成员变量或者函数的基本类型成员变量如果没有初始化会报编译错误，而C++不提供这个机制。Java属于纯面对对象语言，函数只能定义在类里，而C++可以将函数定义在类外。Java函数调用时入参传递，对象类型参数只能传引用不能传值，基本类型参数可以传值，而C++可以使用对象传值方式。Java用带反转域名的类库命名方式来解决名字污染问题，而C++使用名字空间来解决。Java类所有成员函数实现和变量初始化都要写在类定义里，而C++类成员函数在类定义里申明，在类外实现，成员变量可以在类定义里初始化。Java通过类名来引用类的静态成员时，是用类名加.号来引用，而不是C++的:号来引用。Java每个文件只能有一个public类的名称与文件名称相同，包外可以访问这个类但是无法访问其他类，并且这个类有一个main函数，而C++没有这个约束。Java注释采用特定的格式时，可以用javadoc来解析成帮助文档，而C++不提供这个机制。Java不支持操作符重载机制，而C++支持这个机制。Java支持带标签的continue和break语句，用以提供有限的类似C++ goto语句的功能。Java数组不能越界写，否则会报错误，而C++不提供这个机制。Java提供包访问控制权限能让包内的类互相访问，而包外的类不能访问包内的类，而C++不提供这个机制。

以上就是我总结的Java和C++部分不同之处。



javac是用来编译java文件的，javac -d (地址) ().java 表示把java文件编译到指定存储目录下。

java命令是用来执行java编译出的class文件的，java -cp （地址）(包).类名 表示指定class文件的存储目录，执行class文件。

# 写出一段HelloWorld的Java程序，解释程序的各个部分的含义（包定义、包导入、类定义、主函数、打印语句等）。

package hello//包定义，名字随意

public class helloworld {

//public 公开的，class是定义类，后面是名字，和文件名一致

public static void main(String[] args){

//static表示静态的，每个java文件里只允许有一个main类，小括号里的是给主函数传一个字符串

System.out.println("HelloWorld!!!");

//函数，用来输出的

}

}

# 说说自己对Java中基本数据类型、数据类型类的区别和联系的理解

Java提供了两类数据类型：一种是基本数据(原始类型)，一种是引用类型。

基本数据类型的默认值仅在作为类中属性时生效，在方法内部必须先对基本数据类型变量赋值后才能使用，否则编译不通过。Java 是面向对象的语言，但是为了便于开发者的使用，Java 中却沿用了 C 语言的基本数据类型，在进行基本的数据计算时，开发者可以直接使用基础类。但是基本数据类型是不具备对象的特征的，不能调用方法，而且基本数据类型不能存入集合中，所以就需要将基础数据类型实例封装为 Java 对象，使其具有了对象的属性和方法。

## 5、 下面三行代码，哪些是错误的，错误在哪里？

## （1）float income = 300.5f;

没错

## （2）int num = 1.5;

整数型不能直接赋值小数

## （3）if(3){ System.out.println(“测试”);}

If语句中数字不能当true 和 false使用

设计一个函数，返回值是一个长度为n的数组, 数组元素值为1-2n之间的随机数，且不能出现重复。

import java.util.Scanner;

public class ArrayRandom {

public static void main(String[] args){

Scanner reader = new Scanner(System.in);

System.out.println("Enter an integer: ");

int n = reader.nextInt();

int[] array = ArrayCreat(n);

for(int num:array){

System.out.println(num);

}

}

private static int[] ArrayCreat(int n) {

int[] array = new int[n];

retry:

for(int i=0;i<array.length;i++){

int temp=(int)(Math.random()\*n\*2+1);

for(int j=0;j<i;j++){

if(array[j]==temp){

i--;

continue retry;

}

}

array[i]=temp;

}

return array;

}

}