计算机二级JAVA第一章辅导：java程序结构的基本知识 PDF转换可能丢失图片或格式，建议阅读原文 https://www.100test.com/kao_ti2020/181/2021_2022__E8_AE_A1_E7_AE_97_E6_9C_BA_E4_c97_181020.htm 1注释，有三种：//最常用的注释 /* 较长的注释*/ /** 自动生成的文档注释*/ 2数据类型：java有8种原始类型，其中4种整型,2种浮点型，char型，boolean型。所有数值类型都是有符号的.char没有符号，但书中没有提到boolean型是否有符号。 java并非完全面向对象，这8种类型就是非面向对象的，这是因为这8种类型很常用，为了提高程序效率而引用，它们不像java中的其它对象一样是分配在堆中的，而是存储在堆栈中。 注意，类型决定的是行为，而不是存储大小，事实上byte和short的存储都是32位的。不过，数组中的类型是例外，byte型的数组保证每个数组元素只占一个字节。 java为了实现跨平台，在不同处理器下得到可重现的结果，所以严格定义数据类型的长度，如：在任何平台下long总是64位。但这样严格指定会损失性能，还有计算的精度（现在很多平台的long是80位的）。在这种情况下，5.0提供了Strictfp修饰词来修饰类和方法。被在修饰的类或方法进行的计算会严格指定数据类型，否则会根据平台处理器的不同而进行计算。这算是一种折衷吧。 另外，在5.0中char不再使用Unicode编码，而使用UTF-16。当然，UTF-16兼容Unicode。所以这不会对我们的编程产生什么太直观的影响（我是这么认为的）。 我觉得需要注意的一点是：float不适合金融计算，如：2.0 - 1.1得到的结果是0.8999999999999999..因为float型在系统中是二进制表示的，就像十进行中1/3除不尽一样，二进制中的1/10是除不尽的。

其它的没什么好说的了，下面符两张表： Table 3-1. Java Integer TypesTypeStorage RequirementRange (Inclusive)Int4 bytes2,147,483,648 to 2,147,483, 647 (just over 2 billion)Short2 bytes32,768 to 32,767Long8 bytes9,223,372,036,854,775,808 to 9,223,372,036,854,775,807Byte1 byte128 to 127Table 3-2. Floating-Point TypesTypeStorage RequirementRangefloat4 bytesapproximately ±3.40282347E 38F (67 significant decimal digits) double8 bytesapproximately ±1.79769313486231570E 308 (15 significant decimal digits)3操作符： 想一个问题，整数/0会得到什么，浮点数/0又会得到什么？前一个答案是异常，后一个答案是无穷或NaN.浮点型中定义了正无穷，负无穷，和NaN（not a number）三个常量.记住，在和他们进行比较的时候不能用“ == ”，有专门的方法isNaN()等。 java里面没有什么奇特的操作符，大家都应该认识。需要注意的我想就是计算时数据类型间的转换问题 如上图，实线的转换不会丢失数据，虚线的转换可能会丢失数据，沿实线反向转换也是可能会数据。 需要注意操作时会自动转换，原则是： 有double类型的，都转换为double类型； 有float类型的，都转换为float类型； 有long类型的，都转换为long类型； 否则，两个操作数都转换为int类型。 例如，两个byte进行操作，得到是操作结果是int类型的，这点一定要注意。 强制转换： 意味着可能会丢失数据.关键是强制转换后的结果：浮点向整形转换是截取转换，既去掉小数点；大类型向小类型转换,是取模转换，既对小类型的值域取模。 4 Sting: String在java中不是一个原始类型，而是一个类。每个字符串都是这个类的一个实例。 我觉得对于String，应该注意的一点是：String是不可变的。你可

以进行“ ”或substring操作，但最终的结果是产生一个新的String对象，而以前的那个还在。这应该是个考点吧，很多题都和这个相关。对比一个StringBuffer，后者提供了很多对操作其值的方法。操作结果都体现在它自己身上，不会产生另一个StringBuffer对象。 由于上述原因，请注意比较两个字符串是否相等时，不要使用“ == ”，而要使用equle()方法。 ”== ”只能判断两个字符串是否存储在同一个地方. String a = "abc". if(a=="abc"){//很可能返回true} if(a.substring(0,1)=="a"){//很可能返回false} 上述的判断都不能确定。 5 输入/输出 在这里讨论的输入输出，仅仅是讨论简单的对输入的读取和输出的格式化。