

Task1:

一：基本数据类型

1：整形（4）

byte：内存 1 字节；数据范围：-128 ~ 127；

short：内存 2 字节；数据范围：-32768 ~ 32767；

int（默认）：内存 4 字节；数据范围：-2147483648 ~ 2147483647；

long：内存 8 字节；数据范围：-9223372036854775808 ~ 9223372036854775807；

2：浮点型（2）

float：内存 4 字节；数据范围：1.401298 E -45 ~ 3.4028235 E +38；

double：内存 8 字节；数据范围：4.9000000 E -324 ~ 1.797693 E +308；

3：字符型

char：内存 1 字节；数据范围：0 ~ 65535；

4：布尔型

boolean 内存 1 字节；true，false；

二：类型转换

1. 自动类型转化：

定义：类型范围小的类型自动转化为类型范围大的类型

例如 byte short char 可以自动转化到 int

int 可以转化为 long double 等

2. 强制类型转化：

定义：类型范围大的变量强制转化为类型范围小的类型

格式例如

```
Int a =20;
```

```
byte b = (byte) a;
```

即可完成强制类型转化，但责任自负

三．例题说明

是自动类型转化

```
Int a =4;
```

```
char ch ='0';
```

```
int c = a + ch;
```

因为 ch 属于 char 型变量，因而在表达式中会自动转化成 int 类型和 int 类型的 a 相加

而“0” 用二进制表示值是 48，故最后结果为

C = 52

四．包装类，引用数据类型和基本数据类型

1. 包装类：

定义：是一种特殊的类，用于将基本数据类型封装为对象的类

可以使用基本数据类型的方法等

分为 Short Byte Double Integer Character Float Long Boolean;

引用类型：是用于存储对象引用的数据类型，除了基本数据类型之外的其他都叫引用数据类型，其中引用类型变量存储的是对象在内存中的地址，而不是对象本身

基本数据类型缓存池是对基本数据类型的封装类实现的一种内存管理机制，

通过静态内部缓存类来实现一定范围内的数据缓存，在某些范围内的数值，

可通过缓存池来存储，无需每次调用新的对象

该代码输出结果为

true

true

true

原理：Integer x = new Integer(18);

Integer y = new Integer(18);

System.out.println(x == y);

Integer z = Integer.valueOf(18);

Integer k = Integer.valueOf(18);

System.out.println(z == k);

Integer m = Integer.valueOf(300);

Integer p = Integer.valueOf(300);

System.out.println(m == p);

前两行表示新建 Integer 对象，其值为 18，下一行比较 x 和 y 两个对象的引用是否指向内存中的同一个对象；

Integer.valueOf(18);方法，在该方法下再次比较两个对象的引用是否指向同一对象；

五：执行结果

13

6 8

在给 iint 变量 a b 赋值后，++a 先 a 自加后在把 a 赋给 c 而 b 先赋值再自加，故结果为 6+7=13，之后 a=6，b=8；

六：补码：

因为 a =0010，对 a 进行换位取反操作，取反后变为 1101，故-a 的补码为 1110，接下来执行位与操作，只有当两个相应的位都为 1 时，结果位才为 1，因此执行位与操作的所有位都为 0，结果为 0000；

非负整数 a 与 (-a) 进行按位与运算时，因为 java 中负数是按补码的形式，故 a 与 (-a) 结果取决于 a 的二进制形式，

因为 a 二进制中某位为 1，相反位上为 0，而只有两位同时为 1，结果该位才为 1，故 a 与 -a 位与操作后结果通常除了符号位，其他位大多情况下是 0.