**Qingyun Zhang (Rick)**

There is a very obvious difference between the outputs that got in two different programs (call by reference and call by value). In the call by value program, the change made to the variable “a” only vaild inside the addOne() method and had no effect on the original variable “a” that was passed as an argument because the variable “a” was simply copied down into the method as a parameter, and we can treat the “a” in the addOne method as a local variable. Therefore, after executed the addOne method, the variable “a” is still 5. On the other hand, in the call by reference program, changes made to the variable “a” that was a part of the object of the call by reference program which declared before the signature of the main method. The addOne() method directly called the variable “a” from the object and had an effect on the original variable “a” that was passed as an argument, so the result will get 6 instead of 5.

**public** **class** CallByValue {

**public** **static** **void** main(String[] args) {

**int** a=5;

System.out.println("Before calling addOne with a: "+a);

addOne(a);

System.out.println("After calling addOne with a: "+a);

}

**public** **static** **void** addOne(**int** a) {

a++;

}

}

**public** **class** CallByReference {

**public** **int** a;

**public** **static** **void** main(String[] args) {

Number a = **new** Number(5);

System.out.println("Before calling addOne on a: "+a.theNumber);

a.addOne(a);

System.out.println("Before calling addOne on a: "+a.theNumber);

}

}