**DIT636/DAT560 - Mutation Testing Activity**

**The following code iterates over an array and makes all negative values positive.**

1. public int[] makePositive(int[] a){
2. int threshold = 0;
3. for(int i=0; i < a.length; i++){
4. if(a[i] < threshold){
5. a[i]= -a[i];
6. }
7. }
8. return a;
9. }

**1: How many mutations are possible for the following operators:**

* **Relational Operator Replacement**
  + **Swap one of (<,<=,>,>=,==,!=) for one of the others**
* **Arithmetic Operator Replacement**
  + **Swap one of (+, -, \*, /, %) for one of the others.**
  + **Swap one unary (-x, +x) for another**
  + **Swap one shortcut (--x,x--,++x,x++) for another**
  + **Can also swap one unary for one shortcut (e.g., -x to --x)**

**2: Apply the relational operator replacement operation to statement 4 of the method, and identify test input that would lead to a different outcome from the unmutated method. You do not need to create a full unit test.**

**3: Design an equivalent mutant that no test case can detect. You may use any mutation operator discussed in class.**

**4: Design a valid (compiles), but useless (almost all tests will lead to different results than the unmutated method) mutant. You may use any mutation operator discussed in class.**