7AOR:

Definition:

Arithmetic Operator Replacement is defined as the mutation process where one specific operator in an expression is replaced by all other operators.

For example: Let us consider a+b as an expression. This mutation operator should replace the ‘+’ operator by subtraction, multiplication, division and modulus operator.

The possible combinations are:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| + | - | \* | / | % |
| - | \* | / | % | + |
| \* | / | % | + | - |
| / | % | + | - | \* |
| % | + | - | \* | / |

Thought Process:

* Solution 1:
* We need to implement java byte code manipulation to replace these arithmetic operators in the byte code of the program directly.
* We also need to take into consideration the various combinations of operator exchange possible in the system.
* Solution 2: Manually go through the program and replace all of them, which becomes tedious if the program to be tested is really long.

Roadblocks:

* Learning how to access the Java Byte code for that particular method.

Solution/Implementation:

* We took into consideration the data types such as integer, long, double and float for arithmetic operators because each of them are stored differently in the stack.
* We also took into consideration the combinations of operators that can be replaced.
* We used java byte code manipulation to replace these operators directly in the byte code of the program rather than change them manually in the system.
* We used the MethodVisitor function to look through the system and visit every method in the program
* In each method we look for the ADD, SUB, MUL, DIV, REM functions for each respective data type represented in the byte code and replace them with each other using the put function in java
* We maintain HashMap values of each of these and create separate functions for each combination possible to make it faster to implement the test function.

MUTATIONS.put(Opcodes.IADD, new InsnSubstitution(Opcodes.ISUB,"Replaced integer addition with subtraction"));

MUTATIONS.put(Opcodes.ISUB, new InsnSubstitution(Opcodes.IADD,"Replaced integer subtraction with addition"));

MUTATIONS.put(Opcodes.IMUL, new InsnSubstitution(Opcodes.IDIV,"Replaced integer multiplication with division"));

MUTATIONS.put(Opcodes.IDIV, new InsnSubstitution(Opcodes.IMUL,"Replaced integer division with multiplication"));