# **INFO 6205 Final Project Report**

Mengfan Shi Zhengyang Xu

## **Summary**

For this project, our group chose Genetic Algorithms to play the Game of Life. For the initial start pattern, we use Jenetics library to randomly create the first generation pattern

which is a set of bits such as [00011000,10001010,0000000,0000001,00001110,01100000,01100000,00010110].

These pattern bits can be translated into a set of points, in this case the matrix should be like

Then the above points are transferred into eight strings using StringBuilder(). Details can see the InitPattern class.

```
StringBuilder stringBuilder = new StringBuilder();
for (int i = 0; i < gtf.length(); i++) {
   Chromosome<BitGene> bitGenes = gtf.get(i);
   String bit = bitGenes.toString();
   StringBuilder line = new StringBuilder();
   for (int j = 0; j < bit.length(); j++) {
      Character c = bit.charAt(j);
      if (c.equals('1')) {
        line.append(i);
        line.append("");
        line.append(j);
        line.append(",");
   }
}</pre>
```

For the GameStart class, the obtained eight strings should be passed into the method:

```
Game.Behavior run = Game.run(0L, b);
```

By using run.Generation method, we can get the life cycle (generation) of each passed pattern. The survival() method which in our case is the seleceTopPattern() method can be used to select the pattern who has the longest life cycle (generation).

```
private static List<String> selectTopPattern() {
    List<Map.Entry<String, Long>> list = new ArrayList<>(saveGenerationCount.entrySet());
    list.sort((o1, o2) -> (int) (o2.getValue() - o1.getValue()));
    list.forEach(a -> System.out.println(a.getValue() + " " + a.getKey()));
    List<Map.Entry<String, Long>> subList = list.subList(0, 1);
    return subList.stream().map(Map.Entry::getKey).collect(Collectors.toList());
}
```

If the life cycle > 1000, then this pattern is the fittest, output the result, otherwise the pattern should start to mutate, referring to mutation() method until the life cycle is > 1000.

```
<terminated> GameStart [Java Application] /Library/Java/JavaVirtualMachines/jdk-11.jdk/Contents/Home/bin/java (2019
    Group generation: 994
     generation 995; grid=Grid{generation=995, groups=[generation 995, origin = {1, 1}, exter
                              [\{0, 0\}, \{0, 1\}, \{-1, 1\}, \{-2, 0\}, \{-1, 2\}, \{-236, -248\}, \{-237, -249\}, \{-235, -248\}
    generation 995;
     count=12
     Group generation: 995
    generation 996; grid=Grid{generation=996, groups=[generation 996, origin = {1, 1}, exter
                             \{\{0, 0\}, \{0, 1\}, \{0, 2\}, \{-1, 2\}, \{-2, 1\}, \{-236, -248\}, \{-237, -249\}, \{-235, -248\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\}, \{-237, -249\},
    generation 996;
     count=12
     Group generation: 996
    generation 997; grid=Grid{generation=997, groups=[generation 997, origin = {1, 1}, exter
                              [\{0, 0\}, \{-1, 0\}, \{-2, -1\}, \{-1, 1\}, \{-2, 1\}, \{-237, -249\}, \{-238, -250\}, \{-236, -249\}, \{-238, -250\}, \{-238, -250\}, \{-238, -250\}, \{-238, -250\}, \{-238, -249\}, \{-238, -250\}, \{-238, -249\}, \{-238, -250\}, \{-238, -249\}, \{-238, -250\}, \{-238, -249\}, \{-238, -250\}, \{-238, -249\}, \{-238, -250\}, \{-238, -249\}, \{-238, -250\}, \{-238, -249\}, \{-238, -250\}, \{-238, -249\}, \{-238, -250\}, \{-238, -249\}, \{-238, -250\}, \{-238, -249\}, \{-238, -250\}, \{-238, -249\}, \{-238, -250\}, \{-238, -249\}, \{-238, -250\}, \{-238, -249\}, \{-238, -250\}, \{-238, -249\}, \{-238, -250\}, \{-238, -249\}, \{-238, -250\}, \{-238, -249\}, \{-238, -250\}, \{-238, -249\}, \{-238, -250\}, \{-238, -249\}, \{-238, -250\}, \{-238, -250\}, \{-238, -250\}, \{-238, -250\}, \{-238, -250\}, \{-238, -250\}, \{-238, -250\}, \{-238, -250\}, \{-238, -250\}, \{-238, -250\}, \{-238, -250\}, \{-238, -250\}, \{-238, -250\}, \{-238, -250\}, \{-238, -250\}, \{-238, -250\}, \{-238, -250\}, \{-238, -250\}, \{-238, -250\}, \{-238, -250\}, \{-238, -250\}, \{-238, -250\}, \{-238, -250\}, \{-238, -250\}, \{-238, -250\}, \{-238, -250\}, \{-238, -250\}, \{-238, -250\}, \{-238, -250\}, \{-238, -250\}, \{-238, -250\}, \{-238, -250\}, \{-238, -250\}, \{-238, -250\}, \{-238, -250\}, \{-238, -250\}, \{-238, -250\}, \{-238, -250\}, \{-238, -250\}, \{-238, -250\}, \{-238, -250\}, \{-238, -250\}, \{-238, -250\}, \{-238, -250\}, \{-238, -250\}, \{-238, -250\}, \{-238, -250\}, \{-238, -250\}, \{-238, -250\}, \{-238, -250\}, \{-238, -250\}, \{-238, -250\}, \{-238, -250\}, \{-238, -250\}, \{-238, -250\}, \{-238, -250\}, \{-238, -250\}, \{-238, -250\}, \{-238, -250\}, \{-238, -250\}, \{-238, -250\}, \{-238, -250\}, \{-238, -250\}, \{-238, -250\}, \{-238, -250\}, \{-238, -250\}, \{-238, -250\}, \{-238, -250\}, \{-238, -250\}, \{-238, -250\}, \{-238, -250\}, \{-238, -250\}, \{-238, -250\}, \{-238, -250\}, \{-238, -250\}, \{-238, -250\}, \{-238, -250\}, \{-238, -250\}, \{-238, -250\}, \{-238, -250\}, \{-238, -250\}, \{-238, -250\}, \{-238, -250\}, \{-238, -250\}, \{-238, -250\}, \{-238, -250\}, \{-238, -250\}, \{-238, -250\}, \{-238, -250\}, \{-238, -250\}, \{-238, -250\}, \{-238, -250\}, \{-238, -250\}, \{-238, -250\}, \{-238, -250\}, \{-238, -250\}, \{-238, -250\}, \{-238, -250\}, \{-238, -250\}, \{-238, -250\}, \{-238, -25
    generation 997;
     count=12
    Group generation: 997
    generation 998; grid=Grid{generation=998, groups=[generation 998, origin = {1, 1}, exter
                              [\{0, 0\}, \{-1, -1\}, \{0, 1\}, \{-1, 1\}, \{-2, 1\}, \{-237, -249\}, \{-238, -250\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249\}, \{-236, -249
     generation 998;
     count=12
    Group generation: 998
    generation 999; grid=Grid{generation=999, groups=[generation 999, origin = {1, 1}, exter
                              \{\{0, 0\}, \{0, 1\}, \{-1, 1\}, \{-2, 0\}, \{-1, 2\}, \{-237, -249\}, \{-238, -250\}, \{-236, -249\}\}
     generation 999;
     count=12
    Group generation: 999
  Terminating due to: having exceeded 1000 generations
Initial pattern:
```

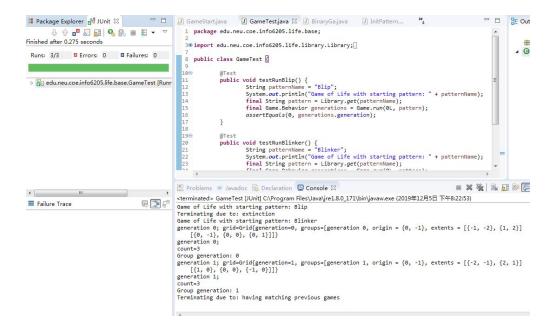
000000000
000000000
First grade pattern:
Second grade pattern
0000000000
000000000
Third grade nettern:
Third grade pattern:
000000000
000000000
Fourth grade pattern:
000000000

```
Fifth grade pattern:
0000000000
0000000000
Methods
  1) Genotype:
[00011000, 10001010, 00000000, 00000001, 00001110, 01100000, 01100000, 00010110]
  2) Phenotype:
0 3,0 4,1 0,1 4,1 6,3 7,4 4,4 5,4 6,5 1,5 2,6 1,6 2,7 3,7 5,7 6
  3) Fitness:
public static boolean fitness(String pattern) {
    return Game.run(0L, pattern).generation >= 1000;
}
  4) Survival:
```

```
private static List<String> selectTopPattern() {
   List<Map.Entry<String, Long>> list = new ArrayList<>(saveGenerationCount.entrySet());
    list.sort((o1, o2) -> (int) (o2.getValue() - o1.getValue()));
    list.forEach(a -> System.out.println(a.getValue() + " " + a.getKey()));
   List<Map.Entry<String, Long>> subList = list.subList(0, 1);
    return subList.stream().map(Map.Entry::getKey).collect(Collectors.toList());
}
  5) Mutate:
public static int[] swapMutation(int[] parent){
    int[] array = parent.clone();
    int l = array.length;
    //get 2 random integers between 0 and the size of the array
    int r1 = randomNumber(0,1);
    int r2 = randomNumber(0,1);
    //to make sure that two numbers are different
    while(r1 == r2) r2 = randomNumber(0,1);
    //swap array elements at those indices
    int temp = array[r1];
    array[r1] = array[r2];
    array[r2] = temp;
    return array;
}
```

### **Unit Tests**

1) GameTest



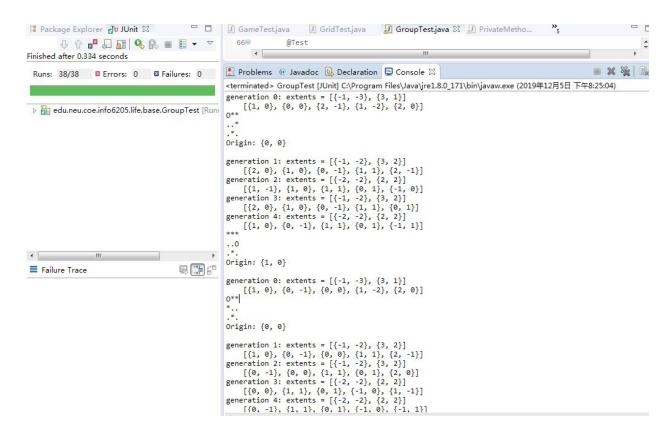
## 2) GridTest

```
🛱 Package Explorer 🗗 JUnit 🛭
                                                   ☑ GameStart.java
                                                                       ☐ GameTest.java ☐ GridTest.java ☒ ☐ PrivateMetho... "5
        // TODO implement test
                                                    26
Finished after 0.148 seconds
                                                     28
 Runs: 7/7 

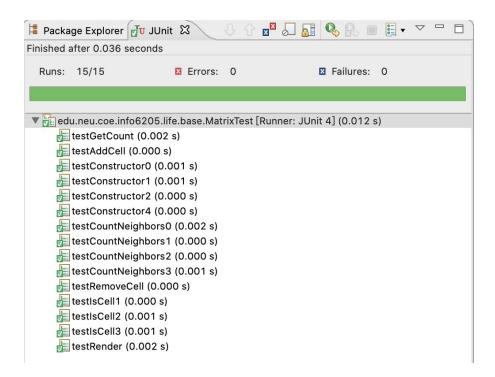
Errors: 0 

Failures: 0
                                                     29⊜
                                                                  public void forEach() {
                                                      30
                                                                           // TODO implement test
                                                                  }
 ▶ 🛅 edu.neu.coe.info6205.life.base.GridTest [Runne
                                                      35
                                                                  public void generation() {
                                                    236
                                                                           // TODO implement test
                                                     38
                                                     396
                                                     40
                                                                  public void testMergeGroups() {
                                                                           Group glider1 = Group.create(0L, Glider1);
Group glider2 = Group.create(0L, Glider2);
                                                     41
                                                     43
                                                                           List<Group> groups = new ArrayList<>();
groups.add(glider1);
                                                     44
                                                                           groups.add(glider2);
                                                    🖹 Problems @ Javadoc 🚇 Declaration 📮 Console 🛭
                                                                                                                                                    ■ × ¾ |
                                                    <terminated > GridTest [JUnit] C:\Program Files\Java\jre1.8.0_171\bin\javaw.exe (2019年12月5日 下午8:24:17)
                                       國 罩 智
Failure Trace
                                                   generation 0, origin = \{0, 0\}, extents = [\{-1, -3\}, \{3, 1\}]
[\{1, 0\}, \{0, 0\}, \{2, -1\}, \{1, -2\}, \{2, 0\}]
```

## 3) GroupTest



#### 4) Matrix Test



## 5) PointTest

```
🖺 Package Explorer 🗗 JUnit 🛭 📅 🗓 GridTestjava 🔑 GroupTestjava 🔑 MatrixTestjava 🗘 PointTestjava
                                                   package edu.neu.coe.info6205.life.base;
       Finished after 0.035 seconds
                                                   3⊕ import edu.neu.coe.info6205.util.PrivateMethodTester;□
 Runs: 13/13 ■ Errors: 0 ■ Failures: 0
                                                      public class PointTest {
                                                              final static Point Origin = new Point(0, 0);
▶ adu.neu.coe.info6205.life.base.PointTest [Runne
                                                  14
                                                              public void getXTest() {
      // TODO implement test
                                               2 15
                                                  17
                                                  186
                                                              public void getYTest() {
                                               20
                                                                      // TODO implement test
                                                              }
                                                  22
23⊜
                                               24
25
26
27
                                                              public void moveTest() {
                                                                      // TODO implement test
                                                              @Test
                                                  28€
                                                              public void testMoveTest() {
                                    國軍部
Failure Trace
                                                  30
                                                                      Point p01 = new Point(0, 1);
Point p10 = new Point(1, 0);
Point p11 = new Point(1, 1);
```

### 6) PrivateMethodTester

```
₽ Package Explorer 🗗 JUnit 🖾
                                            ☑ LibraryTest....
☑ PrivateMetho...
☑ MatrixTest.java
☑ PrivateMetho...
※
                                               2⊕ * Copyright (c) 2018. Phasmid Software
      inished after 0.033 seconds
                                                 package edu.neu.coe.info6205.util;
Runs: 4/4 

Errors: 0 

Failures: 0
                                               7⊕ import org.junit.Test;[]
                                              10
                                              11 public class PrivateMethodTesterTest {
▶ adu.neu.coe.info6205.util.PrivateMethodTester
                                             13⊕
                                                     @SuppressWarnings("SameParameterValue")
                                                     class Mock {
   private final int x;
                                              14
                                                         private final double y;
                                              16
                                            017
                                                         private final String z;
                                             199
                                                         Mock(int x, double y, String z) {
                                             20
                                                             this.x = x;
                                                             this.y = y;
this.z = z;
                                             21
                                             22
                                             23
                                             24
                                            25⊖
                                                         private double xTimesY() {
                                             26
                                                             return x * y;
                                             27
                                             28
                                            29⊖
                                                         private double xTimesYTimesW(Double w) {
                                  國口言
Failure Trace
                                              30
                                                             return x * y * w;
                                              31
                                              32
                                                         nrivate double vTimerVTimerWD]us7/double w int 7\ f
```