

Judgment Of Age By Utilizing Test Suites

Dr. A.Srisaila¹, Dr.T.Srinivasa Ravi Kiran², A. Lakshmanarao³

¹Department of Information Technology, V.R Siddhartha Engineering College, Vijayawada, Andhra Pradesh, PIN: 520007, India

²Department of Computer Science, P.B.Siddhartha College of Arts & Science, Vijayawada, Andhra Pradesh, PIN: 520007, India,

³Department of Computer Science & Engineering, Raghu Engineering College, Visakhapatnam, A.P, India

¹sr.saila@gmail.com, ²kirantsr1@gmail.com, ³laxman1216@gmail.com

Abstract

In particular the purpose of software company is to guarantee production of excessive fantastic software program to stakeholders. Intensive testing with variety of test cases is needed to exercise to make sure high quality of software. At the same time it very important to choose correct test cases with in specified domain all the time when the program being tested. In this paper I make use of Genetic Algorithm (GA) to find the most favourable age of author and his fitness from the details given in the java class file by applying the test cases.

Keywords: *Generic Algorithm, testing, reliability, age, fitness, software.*

1 Introduction:

It is generally stated that the testing approach is an essential part of an effective programming progression process. Testing is to be conducted to identify occurrences of deficiencies, which cause programming failure. On the other hand testing of software is expensive and time consuming. It is proved that the method of checking design specification and source code is a critical section of software improvement process.

A system is to be developed by considering numerous contracts, facts addressed by the customers. Testing necessitate the data that executes the program with typical result.

For the most part, testing procedure contain state box and clear box testing. State box testing is intended to validate the functional requirements where as clearbox testing is completed on syntax, semantics and pragmatics of source code. The amalgamation of state box and clear box testing is called gray box analysis. In general control logic is implemented using decision tree including branches and decision conditions. It is very difficult to test the condition with no branch [1].

By utilizing testing suites testing can be finished manually or automatically. It is found that modernized automated programming testing is better than manual testing.

All things considered, few test data generaion tools are monetarily accessible today [2]. A combination of methods has been proposed to produce test information or test cases without human mediation.

In recent times, majority of tasks are accomplished for the generation of test cases with the practices of soft computing like many valued logic, complex networks and genetic programming. The attribute age is being tested with the control of source code. The test data age must produce positive result irrespective of test suite all the times this phenomenon is called protection of blessed consideration. In this research paper, we formulated a novel strategy to test the consistency of information age using generic algorithm for the given java file.

2 Related Work:

Chayanika Sharma et al. [1] presented a review of special software testing methods where generic algorithm is productively utilized. He also explained selection, crossover or recombination and mutation properties of generic algorithm.

According to Sandeep Goyal et al., the competent test case creation of age with genetic algorithm is more reliable. Implementation of generic algorithms reduces the number of test cases, cost, time and also provide inputs to produce superior software [3].

Deepak kumar et al. deliver a overview on genetic algorithm approach for indiscriminate era of test cases in functional software testing[4].

Xiaoan Bao et al. [5] utilize improved adaptive genetic algorithm (IAGA) for the generation of test cases by keeping up different qualities of population. It apply adaptive mutation rate and crossover rate in adjustment in keeping with the differentiation between personality similarity and robustness values, which improve the operation of penetrating global optimum. The new results prove that the projected technique is capable for generating test cases for coverage of path.

Soniya Malik [6] states that use of generic algorithm leads to minimization of test cases, cost , time and effort to deliver quality software.

3 Proposed Method

In this research article, we followed a novel methodology to test information age of author called as test suite age, and his / her fitness by making use of Generic Algorithm. The source of input is java class file. Test suit will count Hereditary Calculation (HC) of population to find optimal age and fitness of the authors.

Personalized Generic Algorithm:

Present population \leftarrow generate subjective population

Rehash

$K \leftarrow$ first class of present population

While $|K| = | \text{present population} |$ do

$A1, A2 \leftarrow$ Choose two guardians with the determination of rank

On little possibility that hybrid likelihood,

$B1, B2 \leftarrow$ hybrid $A1, A2$

else

$B1, B2 \leftarrow A1, A2$

Change $B1$ and $B2$

$cP = \min(\text{fitness}(A1), \text{fitness}(A2))$ [Select minimum of fitness(A1) and fitness(A2) and store it in cP]

$cO = \min(\text{fitness}(B1), \text{fitness}(B2))$ [Select minimum of fitness(B1) and fitness(B2) and store it in cO]

$dP = \text{length}(A1) + \text{length}(A2)$ [Add the length of A1 and A2 and then store it in dP]

$dO = \text{length}(B1) + \text{length}(B2)$ [Add the length of B1 and B2 and then store it in dO]

EF = superlative individual of current population

On little possibility that $B < cP$ ($cO = cP$ $B1 \leq dP$)

For B in {B1, B2} do

On little possibility that $\text{length}(B) \leq 2 \times \text{length}(EF)$

$K \leftarrow K \cup \{B\}$

else

$K \leftarrow K \cup \{A1 \text{ or } A2\}$

else

$K \leftarrow K \cup \{A1, A2\}$

Present population $\leftarrow K$

The high level description of basic generic algorithm:

Initialize (authors)

Evaluate (authors)

While (stopping condition not satisfied)

{

Selection (Authors)

Crossover (authors)

Mutate (authors)

Evaluate (authors)

}

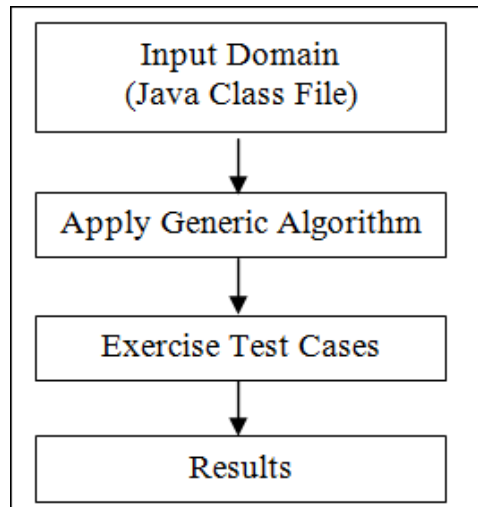


Figure 1. Process flow of customized generic algorithm

4 Results & observations:

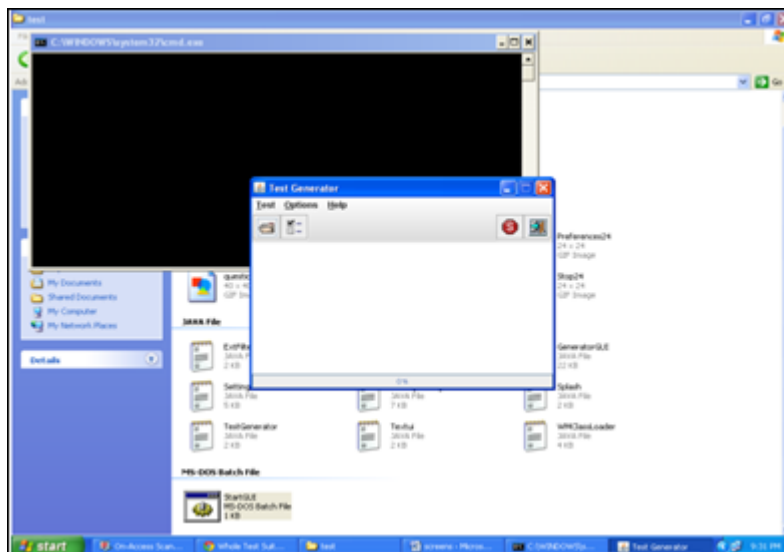


Figure 2. Choose Test Generator

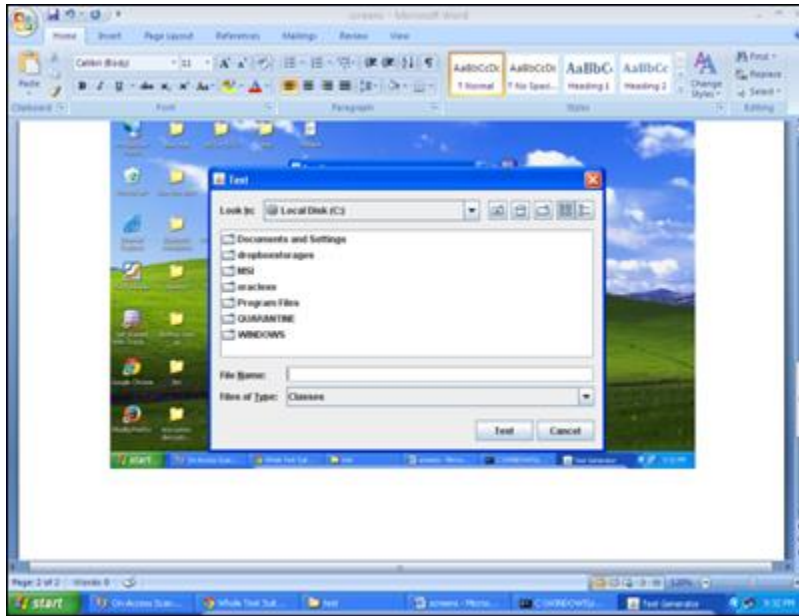


Figure 3. Input Class File

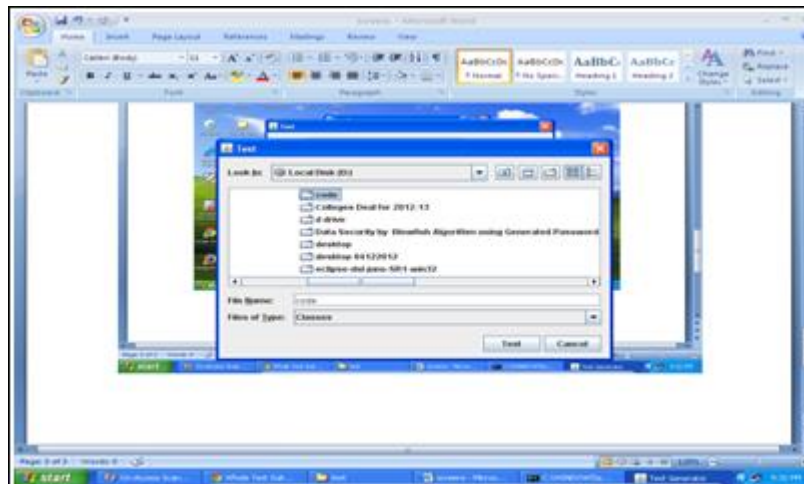


Figure 4. Selection of Code Class

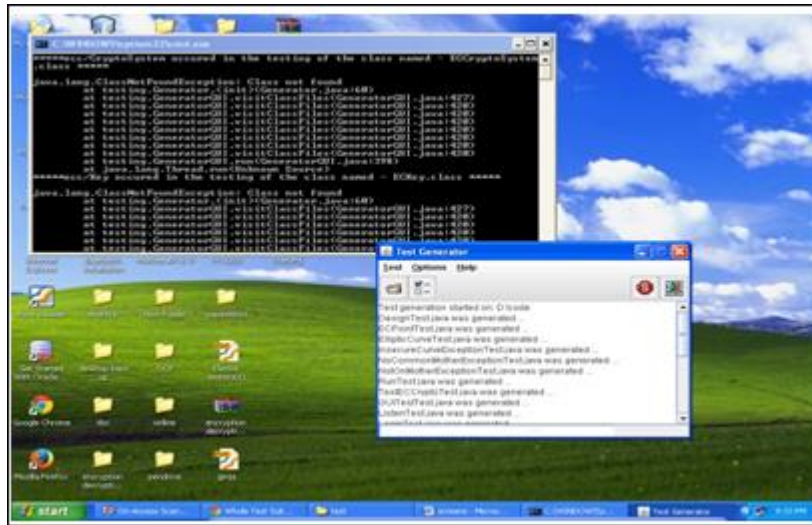


Figure 5. Generation of Design Test

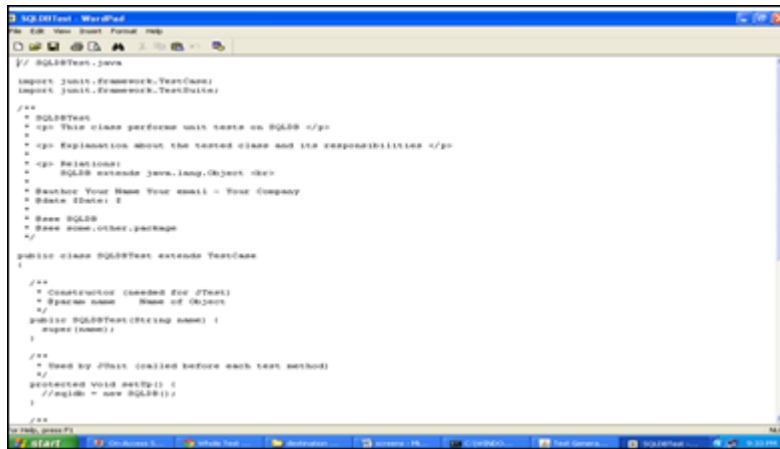


Figure 6. Source code for class SQLDB Test by utilizing Test Case.

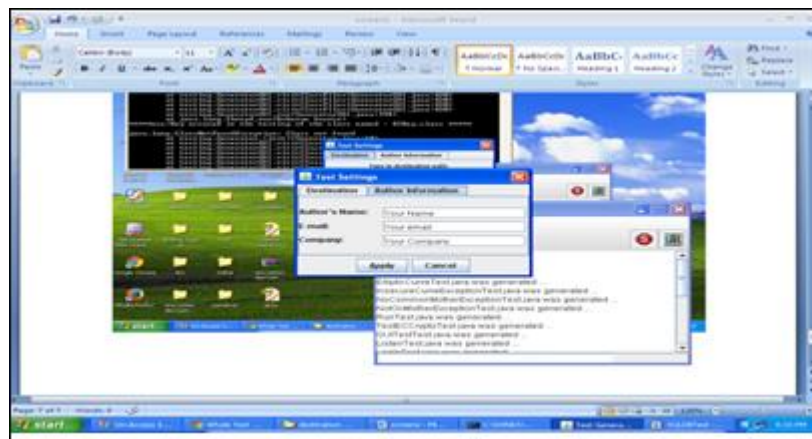


Figure 7. Select Test Settings by means of Creator Data

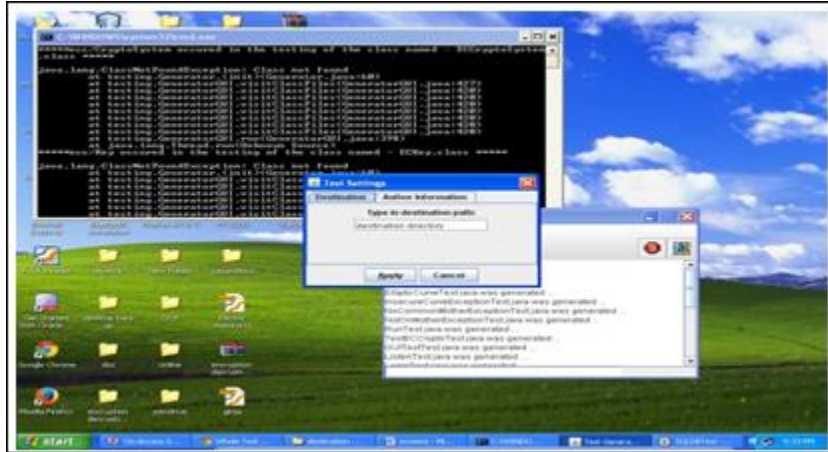


Figure 8. Generation of Test Results of Class to Destination Directory

5 Conclusion:

In this paper, we talk about various appliances of GA in special types of software testing. A schema is defined is to find optimal age of author. The stockholder can analyze find other parameters such health, density, illness, life span of gender and average life.

In future the GA is also be used with computing techniques like fuzzy logic, web applications and in neural networks with distinct types of testing. It is discovered that via using GA, the outcomes and the performance of testing can be improved.

References

1. Chayanika Sharma¹ et al., "A Survey on Software Testing Techniques using Genetic Algorithm", IJCSI International Journal of Computer Science Issues, Vol. 10, Issue 1, No 1, January 2013, ISSN (Print): 1694-0784, ISSN (Online): 1694-0814, www.IJCSI.org
2. Maha alzabidi et al., "Automatic Software Structural Testing by using Evolutionary Algorithms for Test Data Generations", International Journal of Computer science and Network Security, 2009,pp. 390-395.
3. Sandeep Goyal et al., "Software Test Case Optimization Using Genetic Algorithm", International Journal of Scientific Engineering and Science, Volume 1, Issue 12, pp. 69-73, 2018, ISSN (Online): 2456-7361.
4. Deepak Kumar et al., "Genetic Algorithm Approach for Test Case Generation Randomly: A Review", International Journal of Computer Trends and Technology (IJCTT), Volume 49, Number 4, July 2017.
5. Xiaoran Bao et al., "Path-oriented test cases generation based adaptive genetic algorithm", PLOS ONE, <https://doi.org/10.1371/journal.pone.0187471>, November 14, 2017.
6. Soniya Malik, "Software Testing Using Genetic Algorithm", International Conference on Technologies for Sustainability-Engineering, Information Technology, Management and the Environment, DAV Institute of Management, Faridabad, 28th Nov 2015, ISBN:978-81-931039, www.conferenceworld.in.