INTERNATIONAL JOURNAL OF RESEARCH IN COMMERCE, IT & MANAGEMENT

IJRCM

A Monthly Double-Blind Peer Reviewed (Refereed/Juried) Open Access International e-Journal - Included in the International Serial Directories

Indexed & Listed at:

Ulrich's Periodicals Directory ©, ProQuest, U.S.A., EBSCO Publishing, U.S.A., Cabell's Directories of Publishing Opportunities, U.S.A.,

Open J-Gage, India [link of the same is duly available at Inflibnet of University Grants Commission (U.G.C.)],

Index Copernicus Publishers Panel, Poland with IC Value of 5.09 & number of libraries all around the world.

Circulated all over the world & Google has verified that scholars of more than **4600 Cities** in **180 countries/territories** are visiting our journal on regular basis.

Ground Floor, Building No. 1041-C-1, Devi Bhawan Bazar, JAGADHRI – 135 003, Yamunanagar, Haryana, INDIA

http://ijrcm.org.in/

CONTENTS

CALL FOR MANUSCRIPTS

We invite unpublished novel, original, empirical and high quality research work pertaining to recent developments & practices in the areas of Computer Science & Applications; Commerce; Business; Finance; Marketing; Human Resource Management; General Management; Banking; Economics; Tourism Administration & Management; Education; Law; Library & Information Science; Defence & Strategic Studies; Electronic Science; Corporate Governance; Industrial Relations; and emerging paradigms in allied subjects like Accounting; Accounting Information Systems; Accounting Theory & Practice; Auditing; Behavioral Accounting; Behavioral Economics; Corporate Finance; Cost Accounting; Econometrics; Economic Development; Economic History; Financial Institutions & Markets; Financial Services; Fiscal Policy; Government & Non Profit Accounting; Industrial Organization; International Economics & Trade; International Finance; Macro Economics; Micro Economics; Rural Economics; Co-operation; Demography: Development Planning; Development Studies; Applied Economics; Development Economics; Business Economics; Monetary Policy; Public Policy Economics; Real Estate; Regional Economics; Political Science; Continuing Education; Labour Welfare; Philosophy; Psychology; Sociology; Tax Accounting; Advertising & Promotion Management; Management Information Systems (MIS); Business Law; Public Responsibility & Ethics; Communication; Direct Marketing; E-Commerce; Global Business; Health Care Administration; Labour Relations & Human Resource Management; Marketing Research; Marketing Theory & Applications; Non-Profit Organizations; Office Administration/Management; Operations Research/Statistics; Organizational Behavior & Theory; Organizational Development; Production/Operations; International Relations; Human Rights & Duties; Public Administration; Population Studies; Purchasing/Materials Management; Retailing; Sales/Selling; Services; Small Business Entrepreneurship; Strategic Management Policy; Technology/Innovation; Tourism & Hospitality; Transportation Distribution; Algorithms; Artificial Intelligence; Compilers & Translation; Computer Aided Design (CAD); Computer Aided Manufacturing; Computer Graphics; Computer Organization & Architecture; Database Structures & Systems; Discrete Structures; Internet; Management Information Systems; Modeling & Simulation; Neural Systems/Neural Networks; Numerical Analysis/Scientific Computing; Object Oriented Programming; Operating Systems; Programming Languages; Robotics; Symbolic & Formal Logic; Web Design and emerging paradigms in allied subjects.

Anybody can submit the **soft copy** of unpublished novel; original; empirical and high quality **research work/manuscript** ***anytime*** in ***M.S. Word format*** after preparing the same as per our **GUIDELINES FOR SUBMISSION**; at our email address i.e. infoijrcm@gmail.com or online by clicking the link **online submission** as given on our website (***FOR ONLINE SUBMISSION, CLICK HERE***).

GUIDELINES FOR SUBMISSION OF MANUSCRIPT

1. **COVERING LETTER FOR SUBMISSION:**

DATED: _____________

THE EDITOR

IJRCM

Subject: **SUBMISSION OF MANUSCRIPT IN THE AREA OF _______________________.**

(e.g. Finance/Mkt./HRM/General Mgt./Engineering/Economics/Computer/IT/ Education/Psychology/Law/Math/other, please specify)

DEAR SIR/MADAM

Please find my submission of manuscript entitled '____' for possible publication in one of your journals.

I hereby affirm that the contents of this manuscript are original. Furthermore, it has neither been published elsewhere in any language fully or partly, nor is it under review for publication elsewhere.

I affirm that all the co-authors of this manuscript have seen the submitted version of the manuscript and have agreed to their inclusion of names as co-authors.

Also, if my/our manuscript is accepted, I agree to comply with the formalities as given on the website of the journal. The Journal has discretion to publish our contribution in any of its journals.

NAME OF CORRESPONDING AUTHOR :

Designation :

Institution/College/University with full address & Pin Code :

Residential address with Pin Code :

Mobile Number (s) with country ISD code :

Is WhatsApp or Viber active on your above noted Mobile Number (Yes/No) :

Landline Number (s) with country ISD code :

E-mail Address :

Alternate E-mail Address :

Nationality :

NOTES:

a) The whole manuscript has to be in ***ONE MS WORD FILE*** only, which will start from the covering letter, inside the manuscript. ***pdf. version is liable to be rejected without any consideration***.

b) The sender is required to mention the following in the **SUBJECT COLUMN of the mail**:

New Manuscript for Review in the area of (e.g. Finance/Marketing/HRM/General Mgt./Engineering/Economics/Computer/IT/ Education/Psychology/Law/Math/other, please specify)

c) There is no need to give any text in the body of mail, except the cases where the author wishes to give any **specific message** w.r.t. to the manuscript.

d) The total size of the file containing the manuscript is expected to be below **1000 KB**.

e) **Abstract alone will not be considered for review** and the author is required to submit the **complete manuscript** in the first instance.

f) ***The journal gives acknowledgement w.r.t. the receipt of every email within twenty four hours*** and in case of non-receipt of acknowledgment from the journal, w.r.t. the submission of manuscript, within two days of submission, the corresponding author is required to demand for the same by sending a separate mail to the journal.

g) The author (s) name or details should not appear anywhere on the body of the manuscript, except the covering letter and the cover page of the manuscript, in the manner as mentioned in the guidelines.

2. **MANUSCRIPT TITLE:** The title of the paper should be **bold typed**, **centered** and **fully capitalised**.

3. **AUTHOR NAME (S) & AFFILIATIONS:** Author (s) **name**, **designation**, **affiliation** (s), **address**, **mobile/landline number** (s), and **email/alternate email address** should be given underneath the title.

4. **ACKNOWLEDGMENTS:** Acknowledgements can be given to reviewers, guides, funding institutions, etc., if any.

5. **ABSTRACT:** Abstract should be in **fully italicized text**, ranging between **150** to **300 words**. The abstract must be informative and explain the background, aims, methods, results & conclusion in a **SINGLE PARA**. ***Abbreviations must be mentioned in full***.

6. **KEYWORDS:** Abstract must be followed by a list of keywords, subject to the maximum of **five**. These should be arranged in alphabetic order separated by commas and full stop at the end. All words of the keywords, including the first one should be in small letters, except special words e.g. name of the Countries, abbreviations.

7. **JEL CODE:** Provide the appropriate Journal of Economic Literature Classification System code (s). JEL codes are available at www.aeaweb.org/econlit/jelCodes.php, however, mentioning JEL Code is not mandatory.

8. **MANUSCRIPT:** Manuscript must be in ***BRITISH ENGLISH*** prepared on a standard A4 size ***PORTRAIT SETTING PAPER***. *It should be free from any errors i.e.* ***grammatical, spelling*** *or* ***punctuation.*** *It must be thoroughly edited at your end*.

9. **HEADINGS:** All the headings must be bold-faced, aligned left and fully capitalised. Leave a blank line before each heading.

10. **SUB-HEADINGS:** All the sub-headings must be bold-faced, aligned left and fully capitalised.

11. **MAIN TEXT:**

THE MAIN TEXT SHOULD FOLLOW THE FOLLOWING SEQUENCE:

INTRODUCTION

REVIEW OF LITERATURE

NEED/IMPORTANCE OF THE STUDY

STATEMENT OF THE PROBLEM

OBJECTIVES

HYPOTHESIS (ES)

RESEARCH METHODOLOGY

RESULTS & DISCUSSION

FINDINGS

RECOMMENDATIONS/SUGGESTIONS

CONCLUSIONS

LIMITATIONS

SCOPE FOR FURTHER RESEARCH

REFERENCES

APPENDIX/ANNEXURE

The manuscript should preferably range from ***2000 to 5000 WORDS***.

12. **FIGURES & TABLES:** These should be simple, crystal **CLEAR**, **centered**, **separately numbered** & self explained, and **titles must be above the table/figure**. **Sources of data should be mentioned below the table/figure**. *It should be ensured that the tables/figures are referred to from the main text*.

13. **EQUATIONS/FORMULAE:** These should be consecutively numbered in parenthesis, horizontally centered with equation/formulae number placed at the right. The equation editor provided with standard versions of Microsoft Word should be utilised. If any other equation editor is utilised, author must confirm that these equations may be viewed and edited in versions of Microsoft Office that does not have the editor.

14. **ACRONYMS:** These should not be used in the abstract. The use of acronyms is elsewhere is acceptable. Acronyms should be defined on its first use in each section: Reserve Bank of India (RBI). Acronyms should be redefined on first use in subsequent sections.

15. **REFERENCES:** The list of all references should be alphabetically arranged. ***The author (s) should mention only the actually utilised references in the preparation of manuscript*** and they are supposed to follow Harvard Style of Referencing. **Also check to make sure that everything that you are including in the reference section is duly cited in the paper**. The author (s) are supposed to follow the references as per the following:

- All works cited in the text (including sources for tables and figures) should be listed alphabetically.
- Use (**ed.**) for one editor, and (**ed.s)** for multiple editors.
- When listing two or more works by one author, use --- (20xx), such as after Kohl (1997), use --- (2001), etc, in chronologically ascending order.
- Indicate (opening and closing) page numbers for articles in journals and for chapters in books.
- The title of books and journals should be in italics. Double quotation marks are used for titles of journal articles, book chapters, dissertations, reports, working papers, unpublished material, etc.
- For titles in a language other than English, provide an English translation in parenthesis.
- ***Headers, footers, endnotes** and **footnotes** should **not be used** in the document.* However, **you can mention short notes to elucidate some specific point**, which may be placed in number orders after the references.

PLEASE USE THE FOLLOWING FOR STYLE AND PUNCTUATION IN REFERENCES:

BOOKS

- Bowersox, Donald J., Closs, David J., (1996), "Logistical Management." Tata McGraw, Hill, New Delhi.
- Hunker, H.L. and A.J. Wright (1963), "Factors of Industrial Location in Ohio" Ohio State University, Nigeria.

CONTRIBUTIONS TO BOOKS

- Sharma T., Kwatra, G. (2008) Effectiveness of Social Advertising: A Study of Selected Campaigns, Corporate Social Responsibility, Edited by David Crowther & Nicholas Capaldi, Ashgate Research Companion to Corporate Social Responsibility, Chapter 15, pp 287-303.

JOURNAL AND OTHER ARTICLES

- Schemenner, R.W., Huber, J.C. and Cook, R.L. (1987), "Geographic Differences and the Location of New Manufacturing Facilities," Journal of Urban Economics, Vol. 21, No. 1, pp. 83-104.

CONFERENCE PAPERS

- Garg, Sambhav (2011): "Business Ethics" Paper presented at the Annual International Conference for the All India Management Association, New Delhi, India, 19–23

UNPUBLISHED DISSERTATIONS

- Kumar S. (2011): "Customer Value: A Comparative Study of Rural and Urban Customers," Thesis, Kurukshetra University, Kurukshetra.

ONLINE RESOURCES

- Always indicate the date that the source was accessed, as online resources are frequently updated or removed.

WEBSITES

- Garg, Bhavet (2011): Towards a New Gas Policy, Political Weekly, Viewed on January 01, 2012 http://epw.in/user/viewabstract.jsp

IMPACT OF GENETIC ALGORITHM IN CODE COVERAGE FOR TEST SUIT BY JUNIT IN DYNAMIC CONVERSION

SASHIBHUSAN NAYAK
FACULTY
DEPARTMENT OF COMPUTER SCIENCE
RAVENSHAW UNIVERSITY
CUTTACK

ANIL KUMAR BISWAL
ASST. PROFESSOR
DEPARTMENT OF COMPUTER SCIENCE
UDAYANATH (AUTO) COLLEGE OF SCIENCE & TECHNOLOGY
ADASPUR

ABSTRACT

The aim of test coverage suit is to test data which is required for software testing to improving the quality of developed software automatically. Here the automatic testing is essentially needed because the manual operation of test data takes a lot of efforts. That's why automated test data generation is most essentially required. Therefore, to find the suitable optimization technique like Genetic Algorithm (GA) which can be resolved the problem of code coverage for test suit. The genetic algorithm (GA) is generating optimized test suit with the help of Junit tool in different mode of dynamic conversion in test cases. This experiment analyzes the impact of Genetic Algorithm presents how the optimization tool (Junit) generate the optimized test data and minimize the test coverage in dynamic conversion.

KEYWORDS

genetic algorithm, software testing, code coverage, test suit, junit tool.

I. INTRODUCTION

Software testing is a main method for improving the quality and increasing the reliability of software now and thereafter the long-term period future. It is a kind of complex, labor-intensive, and time consuming work; it accounts for approximately 50% of the cost of a software system development. Increasing the degree of automation and the efficiency of software testing certainly can reduce the cost of software design, decrease the time period of software development, and increase the quality of software significantly. Therefore, automatic generation of test data is one of the key research topics in software testing. Today, researchers as well as practitioners use more common methods such as notion to perform, random method and heuristic approaches for test data generation. These methods have some pitfalls in generating test data for larger and complicated programs. So other intelligence techniques have been used very much. The critical point of the problem involved in automation of software testing is of particular relevance of automated software test data generation. Test data generation in software testing is the process of identifying a set of program input data, which satisfies a given testing criterion. For solving this difficult problem, random, symbolic, and dynamic test data generation techniques have been used in the past Software testing is significant because failure in computer software may have severe aftermaths. Software testing is an investigation conducted to provide stakeholders with information about the quality of the product or software under test. Software testing can be stated as the process of validating and verifying that a computer program/application/product:

- meets the requirements that guided its design and development,
- works as expected,
- can be implemented with the same characteristics,
- and satisfies the needs of stakeholders.

Program testing and fault detection can be assisted significantly by testing tools. Testing tools can be put in two classes, static & dynamic.

Static testing involves verification. Static Analyzers probes programs thoroughly and automatically. These are employed on particular language, i.e., these are language dependent. Code Inspectors scrutinizes program to vouch that it hold on minimum quality criteria. Code Inspection activity is found in some COBOL tools (like AORIS librarian system).

Dynamic testing tools involve validation. These are performing analysis of programs on executing them. Coverage Analyzers finds degree of coverage. One of its e.g. Code Coverage tool. Code Coverage Tool is a well-known Eclipse plug-in, employed as white box coverage tool. This tool is very opposite to assure weather TS is giving full code coverage or not. Output Comparators checks weather anticipated and obtained outputs are same or not. JUnit is such a tool. JUnit Tool is Unit Testing framework for Java. It is applied for testing of single component, IT and ST.

Features of Junit are:

- test fixtures for sharing regular test data
- affirmations for testing expected results
- for running tests provides test runners

Static Analyzers and Code Inspectors are static testing tools while Coverage Analyzers and Output Comparators are dynamic testing tools.

II. REQUIREMENT ANALYSIS FOR TEST DATA AUTOMATION

Software testing is a principal technique which is employed for bettering quality attributes of software under test, particularly reliability and correctness.Testing is defined as the process of executing a program with the intent of finding errors. Software should be predictable and consistent, offering no surprises to users. Thus, the problem of test data generation is treated entirely as an optimization problem. The Genetic Algorithms gives most improvements over random testing when these sub domains are small. Experiments show that Genetic Algorithms required less central processing unit (CPU) time in general reaching a global solution than random testing. There are two components to this objective. The first component is to prove that the requirements specification from which the software was designed is correct. The second component is to prove that the design and coding correctly respond to the requirements. Automatic generation of test data helps in reduction of execution time and discovering errors. Automating the process of test data generation reduces the cost in developing test cases.

III. ANALYSIS ON PROBLEM STATEMENT

The furtherance of basic knowledge required to develop new techniques for automatic testing. The main objective is to automate generation of test suit (TS) for each module of SUT by applying GA that could give 100% code coverage.

The performance of Genetic Algorithms in automatically generating test data for small procedures will be assessed and analyzed. A library of Genetic Algorithms will be to apply to large systems. The efficiency of Genetic Algorithms in generating test data will be compared to random testing with regard to the number of test data sets generated and the CPU time required. Coverage analysis is a structural testing technique that helps eliminate gaps in a test suite. It helps most in the absence of a detailed, up-to-date requirements specification. Condition/decision coverage is the best general-purpose metric for C, C++, and Java. Setting an intermediate goal of 100% coverage (of any type) can impede testing productivity. Before releasing, strive for 80%-90% or more coverage of statements, branches, or conditions.

Code coverage analysis is the process of: Finding areas of a program not exercised by a set of test cases,

- Creating additional test cases to increase coverage
- Determining a quantitative measure of code coverage, which is an indirect measure of quality.

An optional aspect of code coverage analysis is:

- Identifying redundant test cases that do not increase coverage.

Test suite use coverage analysis to assure quality of your set of tests, not the quality of the actual product. You do not generally use a coverage analyzer when running your set of tests through your release candidate. Coverage analysis requires access to test program source code and often requires recompiling it with a special command. This paper discusses the details you should consider when planning to add coverage analysis to your test plan. Coverage analysis has certain strengths and weaknesses. You must choose from a range of measurement methods. You should establish a minimum percentage of coverage, to determine when to stop analyzing coverage. Coverage analysis is one of many testing techniques; you should not rely on it alone. Code coverage analysis is sometimes called *test coverage analysis*. The two terms are synonymous. The academic world more often uses the term "test coverage" while practitioners more often use "code coverage". Likewise, a coverage analyzer is sometimes called a *coverage monitor*.

IV. APPROACH OF BASIS PATHS

Our intent is to optimize TS which could give 100 % code coverage. This optimization which is grounded on total code coverage needs that inner composition of program is well-known. Inner composition of program can be discovered by path testing in which a set of test-paths are selected in a program. The different independent paths in the program could be determined through control flow graph (CFG). An independent path is that path in CFG that has one novel set of processing statements or novel conditions. Test cases carrying the information of the path covered by them are grouped together to form initial population of chromosomes and GA is applied. In the end, TS is obtained for each module that gives hundred percent code coverage. The main objective is to develop a test system to exercise all the branches of the software under test. In order to generate the required test data for branch testing genetic algorithms and random testing are used. These two testing techniques will be compared by means of the percentage of coverage which each of them can achieve and by the number of test data.

V. IMPACT OF GENETIC ALGORITHM

GA is an optimization and machine learning algorithm based loosely on the processes of biological evolution. John Holland created the GA field and it is the first major GA publication. GA provides a general-purpose search methodology, which uses the principles of natural evolution. Genetic algorithm as an effective global smart search method, reveals its own strength and efficiency to solve the large space, optimized for high complicated problems, and thus provides a new method to solve the problems of generating test data.

GA solves optimization problems by manipulating initial population (individual chromosomes sampled randomly). Each chromosome is evaluated based on a fitness function which is related to its success in solving a given problem. Given an initial population of chromosomes, GA proceeds by choosing chromosomes to serve as parents and then replacing members of the current population with new chromosomes that are (possibly modified) copies of the parents. The process of selection and population replacement goes on until a stopping criterion (achieving effective test data) has been met. Thus, GA has been successfully used to automate the generation of test data. GA begins with a set of initial population which is randomly sampled for a particular problem domain. Then GA is applied, by performing a set of operations iteratively to get a new and fitter generation.

Generating test data automatically reduces the time and effort of the tester. The two common operations that are performed to produce efficient solution for a target problem after selection operation are Crossover and Mutation. The main idea behind GA is to evolve a population of individuals (candidate solutions for the problem) through competition, mating and mutation, so that the average quality of the population is systematically increased in the direction of the solution of the problem at hand.

The most common operations of a Genetic Algorithm include:

(a) Reproduction: this operation assigns the reproduction probability to each individual based on the output of the fitness function. The individual with a higher ranking is given a greater probability for reproduction. As a result, the fitter individuals are allowed a better survival chance from generation to the next. The selection requires that the solution be evaluated for their fitness as parents: solution that is closer to an optimal solution is judged higher, or fit, than others. After solutions have been evaluated, several are selected in a manner that is biased towards the solutions with higher fitness values. The reason for the bias is that a good solution is assumed to be composed of good component (*genes*). Selecting such solutions as parents increases the chance that their offspring will inherit theses genes and will be at least as fit. Although the selection is biased towards the better solutions, the worst members of the population still have chance of being selected as parents- even a poor solution may have a few good genes that may benefit the population.

(b) *Crossover*: this operation is used to produce the descendants that make up the next generation. This operation involves the following procedures:

(i) select two individuals as a couple from the parent generation.

(ii) randomly select a position of the genes, corresponding to his couple, as a crossover point, thus each individual is divided into two parts.

(iii) exchange the first part of both genes corresponding to the couple.

(iv) add the two resulted individuals to the next generation.

(c) *Mutation*: this operation picks a gene at random and changing its state according to the mutation probability. The purpose of the mutation operation is to maintain the diversity in a generation to prevent premature convergence to local optimal solution. The mutation probability is given intuitively since there is no definite way to determine the mutation probability. Upon completion of the above procedures, a fitness function should be devised to determine which of these parents and offspring's can survive into the next generation. These operations are iterated until the goal is achieved.

VI. METHODOLOGY OF PROPOSED SYSTEM

The concept of GA has been applied to the problem of automated test data generation process. Here the test data is referred to as population in GA. In initial population, each individual bit string (chromosome) is a test data. This set of chromosomes is used to generate test data for feasible basis paths.

The system for generating automated test data for feasible basis paths using GA has been coded in MATLAB. It randomly generates the initial population, evaluates the individual chromosome based on the fitness function value and applies the GA operations such as selection, crossover and mutation to produce next generation. This iterative process stops when the GA finds optimal test data.

The aim of the work is to improve the fitness function as well as to generate the optimal test data. For improving the fitness function of branch predicates, Korel's Distance Function is used. In Korel's distance function, branch predicates are used in the form of relational expression. Using this function, branch predicates are evaluated, as basis path testing includes both statement testing and branch testing. The system for generating automated test data for feasible basis paths using GA has been coded. The basic outline for both algorithms is:

a) Test Data Generation using GA:

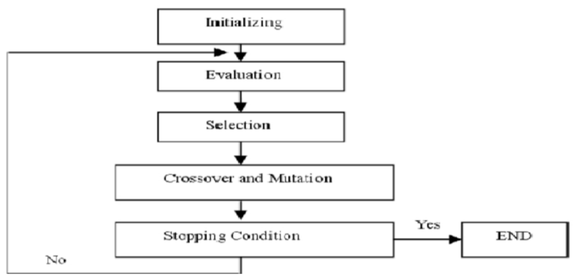
Input: Randomly generated numbers (initial population act as test data) based on the target path to be covered.

Output: Test data for the target path.

i. Gen = 0
ii. While Gen < 100
iii. Do
iv. Evaluate the fitness value of each chromosome based on the objective function.
v. Use roulette wheel as selection operator, to select the individuals to enter into the mating pool.
vi. Perform two-point cross over on the individuals in the mating pool, to generate the new population.
vii. Perform bitwise Mutation on chromosomes of the new population
viii. Gen = Gen +1
ix. go to Step iii.
x. End
xi. Select the chromosome having the best fitness value as the desired result (test data for target path).

b) Test Data Generation using CSA:
i. Gen = 0
ii. Initialize random population A0.
iii. Evaluate Affinity Function An
iv. if Gen > 100 then
v. output= test data
vi. Exit
vii. Else
viii. Clone An to An'
ix. Hyper-mutate An' to An"
x. Evaluate and Select An"
xi. Destroy and renew to construct a new population An
xii. Gen++
xiii. end if
xiv. goto Step iii.

FIGURE 1: STEPS IN GENETIC ALGORITHM



VII. CONVERSION RESULTS

Some Sample problems are taken which are java program. In this module is doing simple task of displaying some statements for analysis of dynamic coverage technique.

FIGURE 2: SAMPLE PROBLEM 1

```
public class test {
public static void main(String args[]){
Scanner s= new Scanner(System.in);
int a=10, b=30;

for(int i=a;i<b;i++){
System.out.println("this is for loop");
}

if(a>b)
System.out.println("this is cfg test again");

for(int i=x;i<y;i++){
System.out.println("this is for loop");
}

if(x>y){
x--;
}

if(y<100){
 y++;
}


if(x<y)
y--;

}
}
```

For the same reason, running the original and the instrumented versions of the Check Value program with JUnit resulted in different code coverage rates, as it can be observed in Figures 3 and 4.

FIG. 3. CHECKVALUE PROGRAM - JUNIT CODE COVERAGE

Element	Coverage	Covered Lines	Missed Lines
CheckValue	75.5 %	37	12
src	75.5 %	37	12
(default package)	75.5 %	37	12
CheckValue.java	70.0 %	28	12
TestEx.java	100.0 %	9	0

FIG. 4: INSTRUMENTED CHECKVALUE PROGRAM - JUNIT CODE COVERAGE

Element	Coverage	Covered Lines	Missed Lines
CheckValue	93.2 %	55	4
src	93.2 %	55	4
(default package)	93.2 %	55	4
CheckValue.java	92.0 %	46	4
TestEx.java	100.0 %	9	0

Since the Junit test cases invoke the program methods directly, running the instrumented CheckValue program with JUnit reached higher coverage rates than running a single test case with JVM. The most important aspect to be noted here is the difference of code coverage for the instrumented and the non-instrumented codes.

VIII. CONCLUSION AND FUTURE WORK

In software testing, the generation of testing data is one of the key steps, which have a great effect on the automation of software testing. Since manual generation of test data consumes much of the computational time, the process of Test Data Generation has been automated. Software Testing is also an optimization problem with the objective that the efforts consumed should be minimized. In this work, optimization of software testing is achieved by employing GA and the process is automated. It results in formulation of test suite for a module that gives 100 % code coverage. The process of code analysis is to find all modules in a program, generation of test suit by junit, determination of all independent paths and GA steps are automated. GA is employed on a set of different software programs and analyses are done on results obtained which decide performance of GA.

Other selection operators and crossover operator can be applied and comparison can be drawn between performances of different operators. In this work very basic fitness function is used. In future, fitness function can be formulated based on *Average Percentage of Condition Coverage* (APCC).

REFERENCES

1. Abdelaziz M. Khamis , Moheb R. Girgis, “Automatic Software Test Data Generation for Spanning Sets Coverage Using Genetic Algorithms”, Vol. 26, pp. 383–401, 2007.
2. Alexandre L. Martins, Simone Hanazumi and Ana C. V. de Melo, “Testing Java Exceptions: an instrumentation technique”, pp. 626-631,2014,IEEE.
3. C. C. Michael, G. E. McGraw, M. A. Schatz, “Generating software test data by evolution”, IEEE Transactions on Software Engineering, vol.27, no.12, pp. 1085-1110, 2001.
4. C. Kaner, J. Falk, and H. Q. Nguyen, Testing Computer Software, 1st ed. New York, NY, USA: Willey Computer Publishing, 1999.
5. Dustin, Elfriede, Jeff Rashka, and John Paul: “Automated software testing” introduction, management, and performance. Addison- Wesley Professional, 1999.
6. JUnit, “A programmer-oriented testing framework for Java,” http://junit.org, accessed: 11/25/2012.
7. Maha Saleh AL-Zabidi, Dr. Ajaya Kumar and Dr. A.D.Shaligram, ”Study of Genetic Algorithm for Automatic Software Test Data Generation”, pp. 65-74, 2013
8. Mohd Athar and Israr Ahmad, “Maximize the Code Coverage for Test Suit by Genetic Algorithm”, pp.431 – 435, 2014.
9. Nirmal Kumar Gupta, Mukesh Kumar Rohil, “Improving GA based Automated Test Data Generation Technique for Object Oriented Software”, pp. 249-253,2013,IEEE.
10. P. Ammann and J. Offutt, Introduction to Software Testing, 1st ed. New York, NY, USA: Cambridge University Press, 2008.
11. Poonam Saini1 and Sanjay Tyagi,” Test Data Generation for Basis Path Testing Using Genetic Algorithm and Clonal Selection Algorithm”, pp. 995-998, 2001.
12. R. P. Pargas, M. J. Harrold, and R. R. Peck, “Test Data Generation using Genetic Algorithms”, Journal of Software Testing, Verifications, and Reliability, vol. 9, pp. 263-282, 1999.
13. Roy P.Pargas, Mary Jean Harrold and Robert R.Peck, “Test Data Generation Using Genetic Algorithm”, pp. 1-19,1999.
14. Yeresime and Suresh Santanu Ku Rath ,“A Genetic Algorithm based Approach for Test Data Generation in Basis Path Testing”, pp. 326-332, 2013.

REQUEST FOR FEEDBACK

Dear Readers

At the very outset, International Journal of Research in Commerce, IT & Management (IJRCM) acknowledges & appreciates your efforts in showing interest in our present issue under your kind perusal.

I would like to request you to supply your critical comments and suggestions about the material published in this issue as well as on the journal as a whole, on our E-mail **infoijrcm@gmail.com** for further improvements in the interest of research.

If you have any queries, please feel free to contact us on our E-mail **infoijrcm@gmail.com**.

I am sure that your feedback and deliberations would make future issues better – a result of our joint effort.

Looking forward an appropriate consideration.

With sincere regards

Thanking you profoundly

Academically yours

Sd/-
Co-ordinator

DISCLAIMER

ABOUT THE JOURNAL

In this age of Commerce, Economics, Computer, I.T. & Management and cut throat competition, a group of intellectuals felt the need to have some platform, where young and budding managers and academicians could express their views and discuss the problems among their peers. This journal was conceived with this noble intention in view. This journal has been introduced to give an opportunity for expressing refined and innovative ideas in this field. It is our humble endeavour to provide a springboard to the upcoming specialists and give a chance to know about the latest in the sphere of research and knowledge. We have taken a small step and we hope that with the active co-operation of like-minded scholars, we shall be able to serve the society with our humble efforts.

Our Other Journals