

# SIMPLE GENETIC ALGORITHM BASED RANDOM TESTING FOR REDUCING FAULTY TEST CASES

<sup>1</sup>D. SREE LAKSHMI, <sup>2</sup>P. BHARAT SIVA VARMA, <sup>3</sup>K. ASHOK KUMAR, <sup>4</sup>DR.Y. SUMANTH, <sup>5</sup>JANJHYAM VENKATA NAGA RAMESH, <sup>6</sup>G. N. SOWJANYA,

<sup>1</sup>Assistant Professor, Department of CSE, PVP Siddhartha Institute of Technology, Vijayawada, A.P, India,

<sup>2</sup>Associate professor, Department of CSE, SRKR Engineering College, Bhimavaram, A.P, India

<sup>3</sup>Assistant Professor, Department of ECE, R.V.R & J.C College of Engineering, Guntur, A.P, India

<sup>4</sup>Associate professor, Department of EEE, R.V.R & J.C College of Engineering, Guntur, A.P, India

<sup>5</sup>Adjunct Professor, Department of CSE, Graphic Era Hill University, Dehradun, India and Department of CSE, Graphic Era Deemed to Be University, Dehradun, Uttarakhand, India

<sup>6</sup>Assistant Professor, Department of CSE, Koneru Lakshmaiah Education Foundation, Vaddeswaram, India

Email: damineni.mtech@gmail.com , pbsvarma@gmail.com, ashok.kavuru@gmail.com, sumanth46@gmail.com , jvnramesh@gmail.com, gnagasowjanya@gmail.com

## ABSTRACT

The goal of software testing is to identify software flaws. Software testing is the process of confirming that a program works as intended. Test inputs are generated at random from the software's input space during random testing. to consistently provide test instances that are random and have some similarities. We will provide a method for minimizing the errors based on the best test cases produced by directed random testing in order to get around these problems. Using the random testing model as a basis, we will create an effective random testing test case in the suggested approach. The Simple Genetic Algorithm (SGA) will be used in this study to create the best inputs, minimizing both equivalent and illicit inputs. SGA makes use of the test case coverage metrics to lessen fault proneness. By merging the old input with the present one, our suggested approach will reduce the input space and improve scalability and efficacy in the software testing age.

**Keywords:** *SGA, Software testing, Coverage, RT*

## 1. INTRODUCTION

Black-box testing is an unsystematic testing technique in which programs are examined by generating autonomous, unsystematic inputs. The key strategy in the middle of several software testing procedures is random testing (RT). Conception is simple. It is often easy to use, has proven effectiveness in detecting malfunctions, and can often figure out the software under test in unexpected ways (Zhou et al., 2012)[1]. Random numbers are frequently used as keys in cryptography. Because key protection depends solely on its number and randomness. For cryptographic applications, random numbers are crucial to generate (Niu et al., 2014)[2].

These requirements are not seriously promised by random number initiators that are heavily involved in corporate operations. The uncertainty, two-dimensionality, and combinatorial huge input gap that these processes must find and train

mechanically, however, is their biggest challenge (Malpani & Bassi, 2012)[3].

According to Tahbaldar and Kalita (2011), the fusion method is utilized to generate time test data in order to compensate for mutually static and dynamic processes.[4]. Because software testing is a very costly process in the software engineering lifecycle, its computerization continues to be a source of great concern (Mcminn, 2013)[5]. A method that identifies progression may need to be developed in order to locate the entity into a specific position, and the group must be launched in order to generate trials that wrap each and every branch in the group (Fraser et al., 2015).

A crucial contribution to improving software eminence is the mechanical production of unit test sets. Techniques such as search-oriented software testing and active representative implementation can effectively produce experiment sets that achieve increased code exposure (Galeotti et al., 2013).[6]. Compared to its traditional documentation, model-oriented scheme testing of

functions through a GUI front-end is more efficient and capable (Darab & Chang, 2014).[7]. Adaptive Random Testing (ART), a kind of RT that is used to increase its efficacy, is used to analyze mathematical programs and is based on break-down models, which include three types: block models, strip models, and point models (Putra & Mursanto, 2013).[8].

However, because of the additional task of ensuring that experiment conditions are dispersed, ART is less effective than random testing. The effectiveness of ART is determined by the amount of time needed to create an experiment condition (Chow et al., 2013)[9]. Random test data generation generates experiment data by allocating unpredictable inputs in a variety of ways. In order to produce experiment data, route-oriented and structural approaches use programs that organize flow diagrams; they select a path and use a procedure, like representative implementation, to produce experiment data. Regardless of the path being used, goal-oriented experiment data generation methods select inputs like reports, condition exposure, and decision exposure to achieve the chosen target (Khan & Nadeem, 2013)[10].

A test collection serves as the source for identification, and often the available experiment collection is not optimal for producing high-quality analytical results. Therefore, creating tests to develop identification is important. There are numerous Search-Based Software Testing (SBST) experiment production techniques now in use. Genetic algorithms are used for global searches (Campos et al., 2012)[11].

## 2. RELATED WORKS

(Phil McMinn et al., 2012)[14] have projected a proper dependence concentrate on resultant from program portion that can be used to keep up with search space diminishing. That record portrays commonly a speculative and trial assessment of the capability of this technique to open source and assembling creation code. The outcomes supply validation to keep up with the case that information field diminishing includes a critical result on the introduction of restricted, around the world, and combination search, while a just irregular hunt is unaltered.

(Andrea Arcur, 2016)[15] has projected examine the obligation that the degree plays in programming testing, especially branch openness. That's what we represent, on "hard" programming testing standard, longer examination series develop their testing

insignificant. In this manner, they debate that the option of the degree of the analysis series was exceptionally critical in programming testing. Speculative review and exploratory investigations on broadly used norm and on exchange programming were achieved to keep up with our cases.

(Barus, Arlinta C et al., 2016)[16] have proposed one more similarity metric to enable multiclass level testing used Workmanship. While making test commitments, they used the likeness metric to register the eliminate between two game plans of articles, and between two progressions of system summons. They coordinate that measurement with Craftsmanship and applied it to a plan of open-source programs, with the trial come about exhibiting that approach beats other RT and Workmanship approaches in OOS testing.

(Syed Abdul Moeed and Niranjana Polala, 2017) [17] investigated the various wellsprings of information which was legitimate or unlawful by using GSA decreases the illegal info and indistinguishable wellsprings of information. Tests were made randomly to avoid the dubiousness and moreover testing was happen in light of GSA health that could lessen the lacks in the framework. The yield result should be legal information that exhibits the issue rate diminishment.

(Chen, Jinfu et al., 2017)[18][19] showed an immediate solicitation Workmanship calculation for programming with non-numeric wellsprings of information. The vital necessities for used Workmanship with non-numeric data sources were a reasonable "eliminate" measure. They used the thoughts of classes and choices from order section testing to detail such an action [20][21]. They look at the mistake acknowledgment apleness of our framework by play out an observational assessment programs, used two standard estimations - F-measure and P-measure. Our Craft calculation quantifiably on a very basic level out performs RT, and shows execution like RT on three of the four lingering projects. The assurance above of our Specialty computation was close to that of R [22][23].

Our fundamental goal in this paper is to create whatever number experiments as would be prudent so that they help uncover however many deficiencies as numerous inclusion focuses as could be allowed. These experiments should be legitimate for each time it creates. Another goal is to build adaptability and adequacy in the time of programming testing while at the same time conveying similar disappointment identification viability

### 3. PROPOSED METHOD

The most essential objective of the projected method is given to the preamble of a skilled cycle for lessen the intelligent shortcoming in keeping through the best trial condition in the immediate irregular testing. To address the adequacy and adaptability of irregular testing t-wise connection deficiencies are thought of. For the standard of developing the investigation conditions the new system proficiently use the Arbitrary Testing. The resultant information sources are conveying to the Straightforward Hereditary Calculation (SGA) when the examination condition foundation ways to deal with a last part. In the Basic Hereditary Calculation, the best data sources are caused what withdraw a broad way in scaling back the denied inputs likewise the vague contributions, in that way dropping the tendency of being issue inclined. The usual methodology of the projected system is effortlessly uncovered in Figure 1. The block representation of the novel strategy is excitingly engraved out in the ensuing fragment

At first, the info application is chosen from the data set for programming testing. Subsequent to choosing the information application the experiments are produced with the assistance of Arbitrary Testing. Here the experiments are created through include esteem from the information application. In the wake of choosing the experiments the bogus decrease is finished by versatile hereditary calculation. In versatile hereditary calculation the ideal data sources are created which will diminish the unlawful data sources and comparable sources of info. In light of the cycle the misleading decrease is finished in the proposed strategy. The clever method courses through the two stages displayed underneath,

1. Test case generation
2. Reducing The Test Cases
  - ❖ Simple Genetic Algorithm

#### Stage 1:

##### 1.1 Test Case Generation

The anticipated technique effectively causes the examination conditions in keeping through the Arbitrary Testing. The capability handled to validation, is viewed as a contribution for the item exercises dependence portrayal in the product testing.

In the movement outline, the gathering of hubs recognizes the items (Ob) and the gathering of edges uncovered the capability (F), where,  $F \in Sf$  recognize the coordinated undertaking, which hold the six trademark far reaching underneath and

incorporate an immediate dependence among the premise and end objects.

Fsource  $\in$  Ob - addresses the wellspring of the capability

Fdest  $\in$  Ob - connects with the objective of the capability and where Fsource  $\neq$  Fdest

Fname - portrayed the name of the capability

FBW  $\in$  Sf - compares to the retrogressive safe capability and where, FBW  $\neq$  F and it is demonstrated as "-".

FER - connotes the probabilistic execution pace of a capability in a Grouping Graph and where,  $0 \leq FER \leq 1$  and the default esteem is one.

FEER - compares to the normal execution pace of a capability in a Grouping Graph and where,  $0 \leq FEER \leq 1$  and the default esteem is one.

We consider the state of a division control design of a premise code, in which the execution charge of an errand is supposed to be impacted. Assume an undertaking is in a change common

segment and just when the prerequisite in the part is fulfilled. In the event that the undertaking is executed encircled by the same detail segment, a while later the chance of execution charge of an errand is 0.5. In any case, the defaulting esteem is one. The expected execution charge of an undertaking represents a chance of the execution charge of a movement graph. Thus, it imparts to the chance of the execution time for the whole measure of undertaking in a careful class to the execution time for the whole measure of assignment in the entire information capability. The utility in a movement chart is completed just when it is prompted. The defaulting worth of FEER is additionally one. Here, every application has the quantity of capability that is utilized for the age of experiment. In light of the capability esteem, the proposed method works out the component esteem..

The proposed technique is primarily center around the capabilities and inclusion measurements of the application that we are utilized for the experiment age. In our executed technique the capability name are address as a variable. In Fig.2 exhibits the overall arrangement of analysis condition development is smoothly uncovered and is proficiently support by unmistakable event exhaustive underneath,

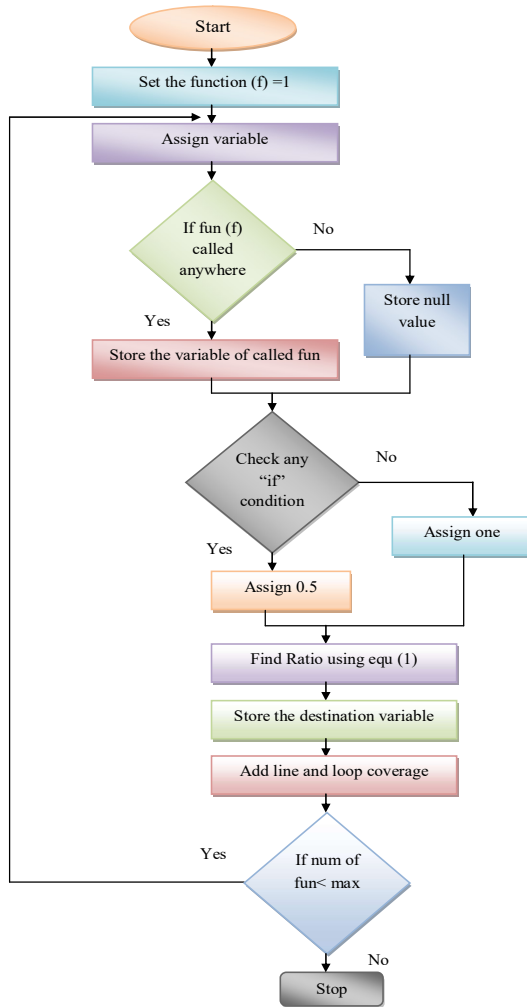


Fig.2 Flowchart For Test Case Generation Using

Figure 2 superbly representation the creation arrangement of the examination condition and every one undertaking is represented as the premise task. Therefore, a premise task is addressing a variable name and afterward each errand is seen to decide if it is recently fixed to distinct beneficial mission. Assuming it is previously apportioned, after that the associated variable name must be conveyed in the analysis condition, if not, it is administered as 'silly'. In the event, any undertaking obtaining the "if" particular, a worth of 0.5 is apportioned in the examination condition; or, more than likely a worth of one must be dispersed. Thusly, the specific assignment extent is assessed followed by the portrayal of the objective errand. At present, the whole analysis conditions are associated with the inclusion grid. In this condition, a chance worth is administered and any errand through the "if" detail is disseminated a worth of 0.5 qualities and an undertaking among no such circumstance is chosen a worth of one. The

Proportion is evaluate for the singular errand according to Condition 1 indicated underneath and hence the objective assignment is recognized, followed by the estimation of the entire examination conditions to the inclusion grid. Proportion esteem is characterized by the proportion of how long one specific capability call the other capability to add up to work. The age utilizing the line inclusion and circle inclusion from the inclusion measurements.

**Line coverage:**

Additionally, line coverage is indicated as a statement that only includes the precise conditions. Additionally, it estimates the code's prominence and ensures that the code in question follows a variety of trails.

**Loop coverage:**

The purpose of the loop coverage tool is to determine and provide light on whether each loop body is executed zero times, precisely once, or several times. Additionally, it indicates whether the loop body is executed exactly once or repeatedly in "do-while" loops. The while-loops and for-loops also generate a lot of presentations. There are no more coverage tools that indicate the comparable data. For a given occasion, we examine the state of a single function that contains two programs and four tasks, each of which is represented by a variable. This particular process is illustrated in the table.

Table I: Sample code

Class A	Class B
A1	B1
IF	{
{	B2
A2	C1
B1	}
}	B2
A2	IF
{	{
C1	A1
}	}

**Examples for line coverage and branch coverage**

**Sample code:**

```

if(cond)
{
line1();
line2();
line3();
line4();
}
else
{
line5();
}
    
```

Four out of five lines and one out of two branches are covered if your test simply tests the condition being true and never executes the else branch. Overall, the percentage of lines that this test run executed is called Line Coverage, and the percentage of branches that this test run executed is called Branch Coverage.

### Examples for test case generation

Variable names for the errands A1, A2, B1, B2 and C1 are conveyed as F1, F2, F3, F4 and F5. . In our proposed strategy, the experiment contains source capability name, likelihood esteem, proportion esteem, objective capability name, line inclusion and circle inclusion. Experiment age process with the comparing model is given underneath,

Experiment 1: [ F1, -, 0.5, 3/6, F2 ] + line inclusion + circle inclusion

Experiment 2: [ F1, -, 0.5, 4/6, F1 ] + line inclusion + circle inclusion

Experiment 3: [ F2, -, 1, 2/6=, F5 ] + line inclusion + circle inclusion

Here, F1 addresses the underlying capability name, 0.5 demonstrates that on the off chance that an experiment contains any in the event that condition or not. In the event that contains implies 0.5 will be relegated. On the off chance that not implies the worth will be 1. Then, at that point, the following worth 2/5 shows the proportion worth of the experiment referenced in condition 1 and F2 addresses the objective capability name. Alongside these qualities, line inclusion and circle inclusion values are utilized for experiment age.

Similarly, all the experiments are produced.

Right now the info task means the financial capability. It incorporates approximately 37 classes and 98 undertakings. The entire measure of trial condition creation in stage 1 is to the abstain of 551. In a vague manner, the total examination conditions are caused, which are thus given to the versatile hereditary calculation. In perception of the truth that the examination conditions are created each time, it is not out of the ordinary encase clear comparability on each example. The first strategy effectively uses the versatile hereditary technique in this way as to diminish the event of issues in the examination conditions.

### Stage 2:

#### 1.2 Test Case Reduction

The Experiment Decrease is on a very basic level portrayed as the limit of extra highlights which are not essential for the rule of regulation. In the feature cycle, a spread of trial condition is created of which a couple are not needing for the regulation mission. Beneficial, there is the presumably of

development of unclear examination conditions at every single time span. Thusly it is all aspects of the extra important to use the administrations of clear effective framework to select the same investigation conditions. In the age making technique, the Versatile hereditary strategy is effortlessly drawn in for the guideline of bogus decrease which is achieved in keeping by the streamlining. It is escort by the method of misleading decrease among the strong guide of the Straightforward Hereditary Calculation (SGA). In the current request, the best data sources are caused according to the Straightforward Hereditary Calculation (AGA) which unequivocally delivers its liability of decreasing the ill-conceived inputs and unclear sources of info. The guide for the comprehensive method of the Versatile Hereditary Calculation is excitingly diagramed in the ensuing section,

#### ❖ Simple Genetic Algorithm (SGA)

For the most part, the Hereditary Calculation represents a creative versatile widespread pursuit strategy captivating sign from the transformative information of the legacy. In this cycle, the Emphasess and the general population are represented as the creation and the chromosomes correspondingly. In perception of the truth that the gathering charge of the traditional GA is lesser, the SGA is productively connected with for essentially surging the gathering charge, through the strong guide of the Cauchy modification as the change engineer, which develop its great outside as the preeminent chosen one in the hereditary calculation for working out power to the GA mission and furthermore to adjust the GA presentation.

SGA recognizes a meta-heuristic technique dedicated to the decrease of the normal improvement framework. It is reliably taken advantage of to incite the clarification to various advancement and explore issue, that techniques initiated by customary turn of events, similar to the legacy, versatile transformation, choice, and hybrid. Strengthening, the contribution of AGA makes the result out of analysis condition development.

#### Initial Phase

The populations of the chromosomes are initially generated at random. symbolizes the population's dimension. The chromosome ( ) encompasses the test cases produced in an arbitrary manner. In this situation, the set of test cases is the chromosome.

#### Fitness Evaluation

Each limitation's fitness value is evaluated, as shown in the following Equation 3, and the chromosome with the highest fitness value is

selected as the best chromosome. For every test case, we have already determined a few factors, like the random test value, fault proneness ratio, line coverage, loop coverage, and so on. Fitness is estimated using these parameters, and the results are shown below.

Here, maximizing fitness to minimize interactive flaws is our challenge. The random test value will present the object behavioural dependency value.

**Selection of Chromosomes**

The "best" chromosomes with the highest fitness value are used to select one or more parent chromosomes, and a creative explanation is produced.

**Transition**

At the crossover charge of (), the single peak crossover is applied, and as a result, (") issues are resolved. Genes are transferred between the comparative parents in the context of every crossover task.

**Mutation**

Transformation and Hybrid are the two significant hereditary activity which will help for arrangement union. Here the versatile way of behaving of GA will be introduced with regards to Versatile change.

Here we are involving Cauchy's transformation for adaptiveness in GA. The substances are concerned probabilistically to convey a revolutionary change in themselves. Though using the change engineer, there is plausible that unequivocal unique characteristics arise on depiction of the adjustment in the chromosome. The Cauchy change is effectively taken advantage of to adjust the elements according to Condition 4 uncovered. The change is executed in keeping by the fixed changing chance. Where the Cauchy modification is accomplished, unreasonable variable 'x' represent a Cauchy portion. The Cauchy distribution task is unmistakable according to the ensuing Condition 4,

$$F(x) = \frac{1}{2} + \frac{1}{\pi} \arctan(x) \tag{4}$$

Here x represents the test cases.

**Updating of solution**

Following the completion of the transformation task, an inventive chromosome is created, which then surrogates the current chromosome. The modernization of explanation is the name given to this system. It is the creative chromosome that is

chosen as the best chromosome if its fitness value is higher than the existing one.

**Termination Criteria**

The process continues until the termination requirements are met. Figure 3 displays the unique technique's flowchart in an elegant manner.

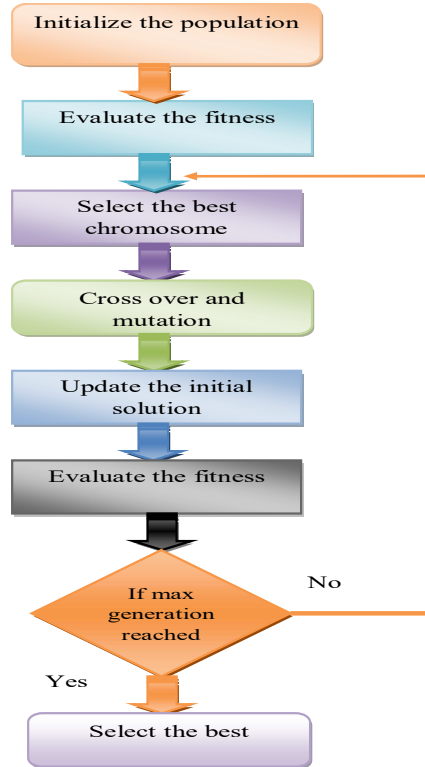


Fig.3 Flowchart For The Proposed Simple Genetic Algorithm

The most great item is differentiated by the total errand resulting to achieving the end from the versatile hereditary calculation. Some errand which doesn't surrender the best outcome must be promptly withdrawn from the capability. Consequently the versatile hereditary calculation is skilled to achieve significant diminishing in the slip-up charge depend on the best examination condition.

In the creative strategy, the undertaking of the SGA is the choice of the best ideal trial conditions which are appropriate for the information capability. Subsequently, the fake reducing is achieved through the improvement calculation and the advanced end is accomplished depend on the wellness esteem chosen the SGA interaction. The examination conditions gave to the improvement

calculation are drilled in keeping through the wellness values. The examination conditions enveloping better wellness values connote the bug free capability. In this manner, through the guide of the SGA system, the fundamental analysis conditions are achieved for the creative technique.

**IV Result and Discussion**

We employed the adaptive genetic algorithm for reduction in our experiment. Interactive Errors in Directed Random Testing Based on Ideal Test Cases. The following are the outcomes of the implementation, which was carried out on the Java platform.

The entire procedure in a genetic algorithm is based on the fitness value of the selected parameter. The parameter with the highest fitness value is selected for further processing. The chromosomes' fitness values for distinct iterations are calculated for the Adaptive Genetic Algorithm, and the outcomes are plotted. Table 2 shows the fitness value for the recommended method, which is confirmed to be better than the method that uses GA.

**Table 2:** Fitness comparison for different iterations using our proposed SGA and GA.

Iterations	Fitness values		
	Simple Genetic Algorithm	Genetic Algorithm	PSO
25	631	567	621
50	611	530	610
75	583	485	558
100	556	465	496

The number of test cases derived from the various current approaches and our suggested SGA method is displayed in table 3 below. Compared to other approaches, we have a higher number of test cases.

*Table 3: Test case count for proposed and existing methods*

Iterations	Test case count		
	SGA	GA	PSO
25	455	485	499
50	477	490	496
75	457	481	492
100	434	492	499

The time and memory consumption of our suggested methods are displayed in table 4 below. The associated time and memory usage are computed for every iteration, and the outcomes are tabulated. We decrease the execution time and memory use by lowering the interactive faults in this case. Time and memory use automatically decrease as the number of iterations increases.

*Table 4: Time and Memory usage for different iterations using proposed SGA.*

Iteration	Time usage (sec)	Memory usage (Bytes)
25	5367	5697247
50	5400	4112399
75	5636	3084790
100	5613	3299423

The time consumption derived from the various current approaches and our suggested AGA method is displayed in table 5 below. Compared to previous approaches, we have achieved superior time usage. The difference between the process start and finish system times is used to compute the amount of time used. Below is a tabulation of table 5.

Table 5: Comparison Of Time Usage In Proposed And Existing Method.

Iteration	Time usage (sec)		
	SGA	GA	PSO
25	5367	5632	6325
50	5400	5709	6476
75	4636	5835	6433
100	5613	5803	6852

Compared to other tests, such as regression testing, we find that interactive errors are more easily found using random testing. RT will produce better results if the test suite is large. The suggested technique's real-time application is utilized to examine the suitability of Combinatorial Interaction Testing (CIT) in a fresh way. Thousands or even hundreds of features have been examined in real-world industrial systems.

## 5. CONCLUSION

In the article, an imaginative method for decrease the intelligent error is actuated which dependent on best examination conditions in the expected for unsystematic is explore. The first strategy is locked in to decrease the intuitive slip-ups according to the versatile hereditary calculation. The versatile hereditary calculation builds the best item which altogether decreases the disallowed inputs. By an eye on decline the imperfections; the first technique effectively uses the inclusion measurements. The uplifting item uncover the truth that the imaginative method arises by flying tones in getting free of the vagary of emotionally shaped explore conditions and delivering the best item. In the days to show up, it is likely to perceive the analysis condition distribution measurements as trial condition decision standard for achieving the more noteworthy inclusion try conditions rapidly. Valuable, unmistakable further imaginative meta-heuristic examine utensils will be taken advantage of to cause the imperfection free trial conditions.

## REFERENCES

- [1] Zhou, Z. Q., Sinaga, A., & Susilo, W. (2012, January). On the fault-detection capabilities of adaptive random test case prioritization: Case studies with large test suites. In System Science (HICSS), 2012 45th Hawaii International Conference on (pp. 5584-5593). IEEE.
- [2] Niu, X., Wang, Y., & Wu, D. (2014, August). A Method to Generate Random Number for Cryptographic Application. In Intelligent Information Hiding and Multimedia Signal Processing (IIH-MSP), 2014 Tenth International Conference on (pp. 235-238). IEEE.
- [3] Dionísio, J., Mota, T., Pinto, I., & Niehus, M. (2014). Real Time Random Number Generator Testing. Procedia Technology, 17, 534-541.
- [4] Malpani, P., & Bassi, P. (2014, September). Analytical & empirical analysis of external sorting algorithms. In Data Mining and Intelligent Computing (ICDMIC), 2014 International Conference on (pp. 1-6). IEEE.
- [8] Galeotti, J. P., Fraser, G., & Arcuri, A. (2013, November). Improving search-based test suite generation with dynamic symbolic execution. In Software Reliability Engineering (ISSRE), 2013 IEEE 24th International Symposium on (pp. 360-369). IEEE.
- [9] Darab, M. A. D., & Chang, C. K. (2014, October). Black-box test data generation for gui testing. In Quality Software (QSIC), 2014 14th International Conference on (pp. 133-138). IEEE.
- [10] Putra, I. P. E. S., & Mursanto, P. (2013, September). Centroid Based Adaptive Random Testing for object oriented program. In Advanced Computer Science and Information Systems (ICACISIS), 2013 International Conference on (pp. 39-45). IEEE.
- [11] Chow, C., Chen, T. Y., & Tse, T. H. (2013, July). The art of divide and conquer: An innovative approach to improving the efficiency of adaptive random testing. In Quality Software (QSIC), 2013 13th International Conference on (pp. 268-275). IEEE.
- [12] Khan, S. A., & Nadeem, A. (2013, April). Automated test data generation for coupling based integration testing of object oriented programs using evolutionary approaches. In Information Technology: New Generations



- (ITNG), 2013 Tenth International Conference on (pp. 369-374). IEEE.
- [13] Campos, J., Abreu, R., Fraser, G., & d'Amorim, M. (2013, November). Entropy-based test generation for improved fault localization. In Proceedings of the 28th IEEE/ACM International Conference on Automated Software Engineering (pp. 257-267). IEEE Press.
- [14] Arcuri, A., & Briand, L. (2012). Formal analysis of the probability of interaction fault detection using random testing. IEEE Transactions on Software Engineering, 38(5), 1088-1099.
- [15] Arcuri, A. (2012). A theoretical and empirical analysis of the role of test sequence length in software testing for structural coverage. IEEE Transactions on Software Engineering, 38(3), 497-519.
- [16] Kempka, J., McMinn, P., & Sudholt, D. (2013, July). A theoretical runtime and empirical analysis of different alternating variable searches for search-based testing. In Proceedings of the 15th annual conference on Genetic and evolutionary computation (pp. 1445-1452). ACM.
- [17] Arts, T., Gerdes, A., & Kronqvist, M. (2013, September). Requirements on automatically generated random test cases. In Computer Science and Information Systems (FedCSIS), 2013 Federated Conference on (pp. 1347-1354). IEEE.
- [18] Barus, A. C., Chen, T. Y., Kuo, F. C., Liu, H., Merkel, R., & Rothermel, G. (2016). A cost-effective random testing method for programs with non-numeric inputs. IEEE Transactions on Computers, 65(12), 3509-3523.
- [19] Syed Abdul Moeed and Niranjana Polala, (2017) "Interactive Faults Detection (Ifd) In Acceptance Testing Based On Optimal Test Cases Using Gravitational Search Algorithm (Gsa)," International Journal of Pure and Applied Mathematics Volume 115( 8), 559-564.
- [20] Chen, J., Kuo, F. C., Chen, T. Y., Towey, D., Su, C., & Huang, R. (2017). A Similarity Metric for the Inputs of OO Programs and Its Application in Adaptive Random Testing. IEEE Transactions on Reliability, 66(2), 373-402.
- [21] Praveen, S. P., Chokka, A., Sarala, P., Nakka, R., Chandolu, S. B., & Jyothi, V. E. (2024). Investigating the Efficacy of Deep Reinforcement Learning Models in Detecting and Mitigating Cyber-attacks: a Novel Approach. Journal of Cybersecurity & Information Management, 14(1).
- [22] Praveen, S. P., Bikku, T., Muthukumar, P., Sandeep, K., Sekhar, J. C., & Pratap, V. K. (2024). Enhanced Intrusion Detection Using Stacked FT-Transformer Architecture. Journal of Cybersecurity & Information Management, 13(2).
- [23] Bikku, T., Chandolu, S. B., Praveen, S. P., Tirumalasetti, N. R., Swathi, K., & Sirisha, U. (2024). Enhancing Real-Time Malware Analysis with Quantum Neural Networks. Journal of Intelligent Systems and Internet of Things, 12(1), 57-7.