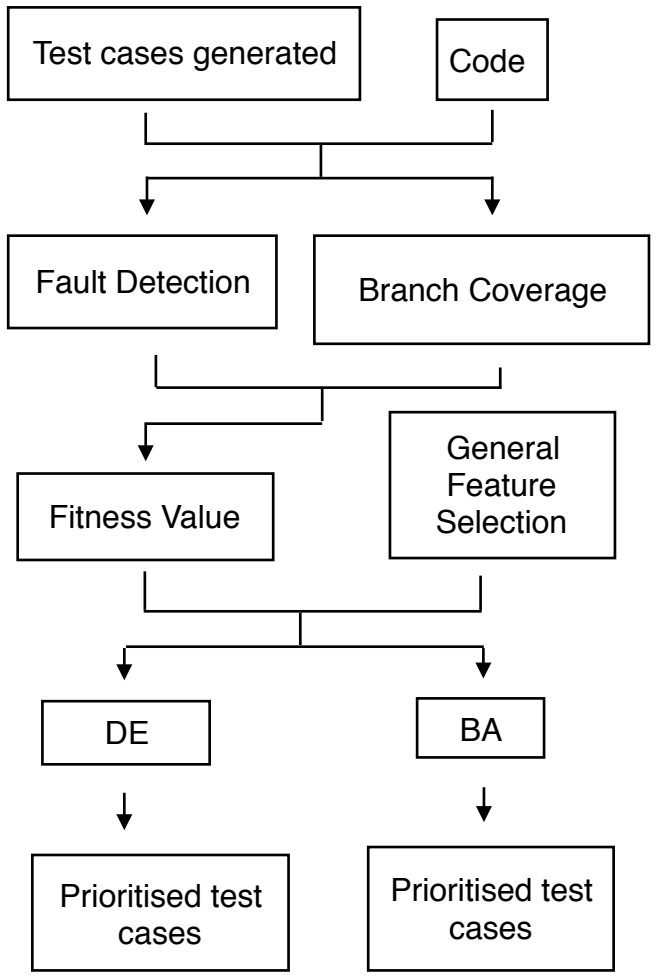
JAYPEE INSTITUTE OF INFORMATION TECHNOLOGY
DEPARTMENT OF
COMPUTER SCIENCE AND ENGINEERING

TEST CASE OPTIMISATION USING NATURE INSPIRED ALGORITHM

Test case optimisation deals with selecting effective test cases having maximum code coverage and fault detection capability, consequently minimising and prioritising the test cases to reduce the cost and timing of testing process.

To Research about Test Case generation, optimisation and prioritisation using Nature inspired algorithms like Bacteriologic Algorithm, Differential Evolution Algorithm and hence create a tool for comparison amongst the above mentioned algorithm with respect to test case prioritisation and optimisation.

Programs	Differential Evolution	Bacteriologic Algorithm
Leap Year	0.09099	0.11807
Maximum of three	0,07788	0.07906
Minimum of two	0.07181	0.12891
Point Circle	0.11112	0.096137
Remainder	0.05966	0.11210
Quadretic Eqn.	0.04116	0.08010
Even Odd	0.04732	0.05836
Marks	0.07820	0.08265
Triangle clas.	0.08925	0.10104



	DIFFERENTIAL EVOLUTION ALGORITHM	BACTERIOLOGIC ALGORITHM
1. Begin	Generate randomly an initial population of the solution.	Generate randomly an initial population of the solution.
	Calculate the fitness value of the initial population.	Calculate the fitness value of the initial population.
2. Repetition	For each parent, select three solutions at random.	Select a pair of parents on the basis of the fitness value.
	Create one off-spring using DE operators.	Create two off-springs using crossover.
	Performs above step for the number of times equal to the population size.	Apply mutation to each child.
	For each member of the next generation (i) If off-spring(x) is more fit than the parent(x) (ii) Parent(x) is replaced.	Evaluate the mutated off-spring
	Continue to work until stop is condition is satisfied.	All the off-springs will be the new population, the parents will die.
		Continue to work until stop is condition is satisfied.

Both algorithm's have performed well in test case prioritisation and optimisation but we can see that Differential Evolution Algorithm is more efficient in terms of time for the stated problems.

SUBMITTED BY -

VIBHOR GUPTA 9913103515

NAMAN SHARMA 9913103546

AVNEET SINGH 9913103676