Genetic Algorithm

Genetic algorithm:

Genetic algorithm is a global search technique that works in discrete space. It's random search algorithm but converge.

Phyno Type:

The phyno type is the Actual search space.

Geno Type:

Geno type is the representation search space.

Population:

is a subset of solutions in the current generation. It can also be defined as a set of chromosomes.

Gene:

An individual is characterized by a set of parameters (variables) known as **Genes**. Genes are joined into a string to form a **Chromosome**

Chromosome:

is a set of parameters which define a proposed solution to the problem that the genetic algorithm is trying to solve.

Chromosome

1	2	3	4	5	6	7
A	b	c	d	e	f	g

Gene

4
d

Allele

d

LOLOS

4

Five phases are considered in a genetic algorithm.

1. Initial population
2. Fitness function
3. Selection
4. Crossover
5. Mutation

Initial Population:

This process begins with a set of individuals which is called a **Population**. Each individual is a solution to the problem you want to solve.

Fitness function:

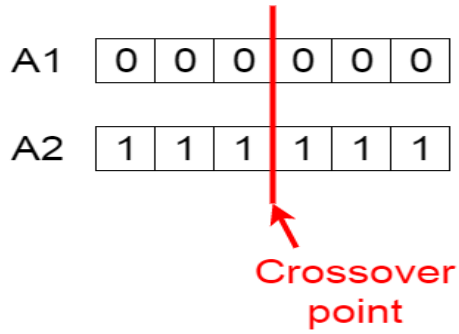
The **fitness function** determines how fit an individual is (the ability of an individual to compete with other individuals).

Selection

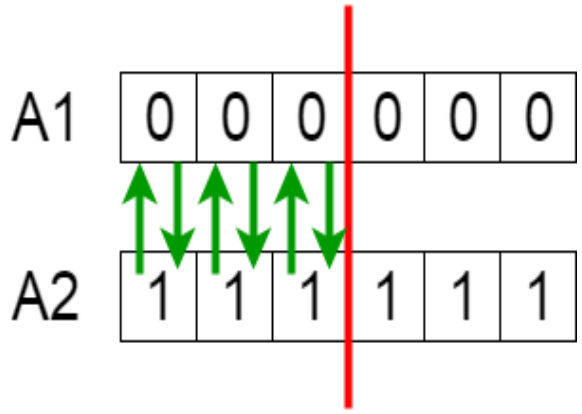
The idea of **selection** phase is to select the fittest individuals and let them pass their genes to the next generation.

Cross over:

Crossover is the most significant phase in a genetic algorithm. For each pair of parents to be mated, a **crossover point** is chosen at random from within the genes.



Offspring are created by exchanging the genes of parents among themselves until the crossover point is reached.



The new offspring are added to the population.

A5	1	1	1	0	0	0

A6	0	0	0	1	1	1

Mutation:

In certain new offspring formed, some of their genes can be subjected to a **mutation** with a low random probability.

This implies that some of the bits in the bit string can be flipped.

Before Mutation

A5	1	1	1	0	0	0

After Mutation

A5	1	1	0	1	1	0

Natural Selection:

The process of natural selection starts with the selection of fittest individuals from a population. They produce offspring which inherit the characteristics of the parents and will be added to the next generation. If parents have better fitness, their offspring will be better than parents and have a better chance at surviving. This process keeps on iterating and at the end, a generation with the fittest individuals will be found.

Pseudocode:

- Choose the initial population of individuals
- Evaluate the fitness of each individual in population
- Repeat until termination condition satisfied:
  - Selection: Select the individuals with greater fitness for reproduction
  - Crossover: Breed new individuals through crossover
  - Mutation: Apply probabilistic mutation on new individuals
  - Form a new population with these offsprings.
- Terminate

Maximize $F(x) = X^2$ $0 \leq x \leq 31$

- population size = 6;

- Crossover rate = 0.6 : 0.9

- mutation rate = 1 / (population size * chromosomes lenght)

➔ Solution ::

i	X_i	chromosomes	Fitness_func_i	Crossover	Mutation
1	13	[0,1,1,0,1]	169	[0,1,1,0,1]	[1,1,1,0,1]
2	1	[0,0,0,0,1]	1	[0,0,0,0,1]	[0,0,0,0,1]
3	9	[0,1,0,0,1]	81	[0,1,1,1,1]	[0,1,1,1,1]
4	7	[0,0,1,1,1]	49	[0,0,0,0,1]	[0,0,0,0,1]
5	5	[0,0,1,0,1]	25	[0,0,1,0,0]	[0,0,1,0,0]
6	12	[0,1,1,0,0]	144	[0,1,1,0,1]	[0,1,1,0,1]
Total(sum)			469		
Average			78.16		
Max			169		

➔ determining the position of the crossover point. This is done by generating random numbers between 1 to (length of Chromosome – 1).
In this case, generated random numbers should be between 1 and 4. After we get the
crossover point, parents Chromosome will be cut at crossover point and its gens will be
interchanged. ANS:[2,3,3]

→ generate random number [0 -> 1] if number more than 0.9 [crossover rate] then don't apply crossover otherwise apply crossover.

→ in this problem select [(1,2) , (3,4) , (5,6)] to apply crossover.

→ mutation rate = 1/(6*5) = .033

➔ next iteration:

i	X_i	Fitness_ func_i	Fitness_func_i_%	X_i	chromosome	Fitness _func_i	Crossover	Mutation
1	29	841	0.67	29	[1,1,1,0,1]	841	[1,1,1,1,1]	[1,1,0,1,1]
2	1	1	0.0001	15	[0,1,1,1,1]	225	[0,1,1,0,1]	[0,1,1,0,1]
3	15	225	0.17	29	[1,1,1,0,1]	841	[1,1,1,0,0]	[1,1,1,0,0]
4	1	1	0.0001	4	[0,0,1,0,0]	16	[0,0,1,0,1]	[0,0,1,0,1]
5	4	16	0.012	29	[1,1,1,0,1]	841	[1,1,1,1,1]	[1,1,1,1,1]
6	13	169	0.13	15	[0,1,1,1,1]	225	[0,1,1,0,1]	[0,1,1,0,1]
T		1253						
		208						

→ in this iteration we use roulette wheel to select next generation

→ select mating pool by spinning roulette wheel 6 times.

→ generate random number and multiply in Fitness_func_i_% if more than 50% select chromosome otherwise no selection;

→ the position of crossover points [2,3,3];

→ parents to apply crossover [(1,2),(3,4),(5,6)];

→ in mutation select one gene Random to make change in it;