
GENETIC ALGORITHM
thymine
guanine
adenine

GENETIC ALGORITHM

INTRODUCTION

- Genetic Algorithm (GA) is a search-based optimization technique based on the principles of Genetics and Natural Selection. It is frequently used to find optimal or near-optimal solutions to difficult problems which otherwise would take a lifetime to solve.

INTRODUCTION TO OPTIMIZATION

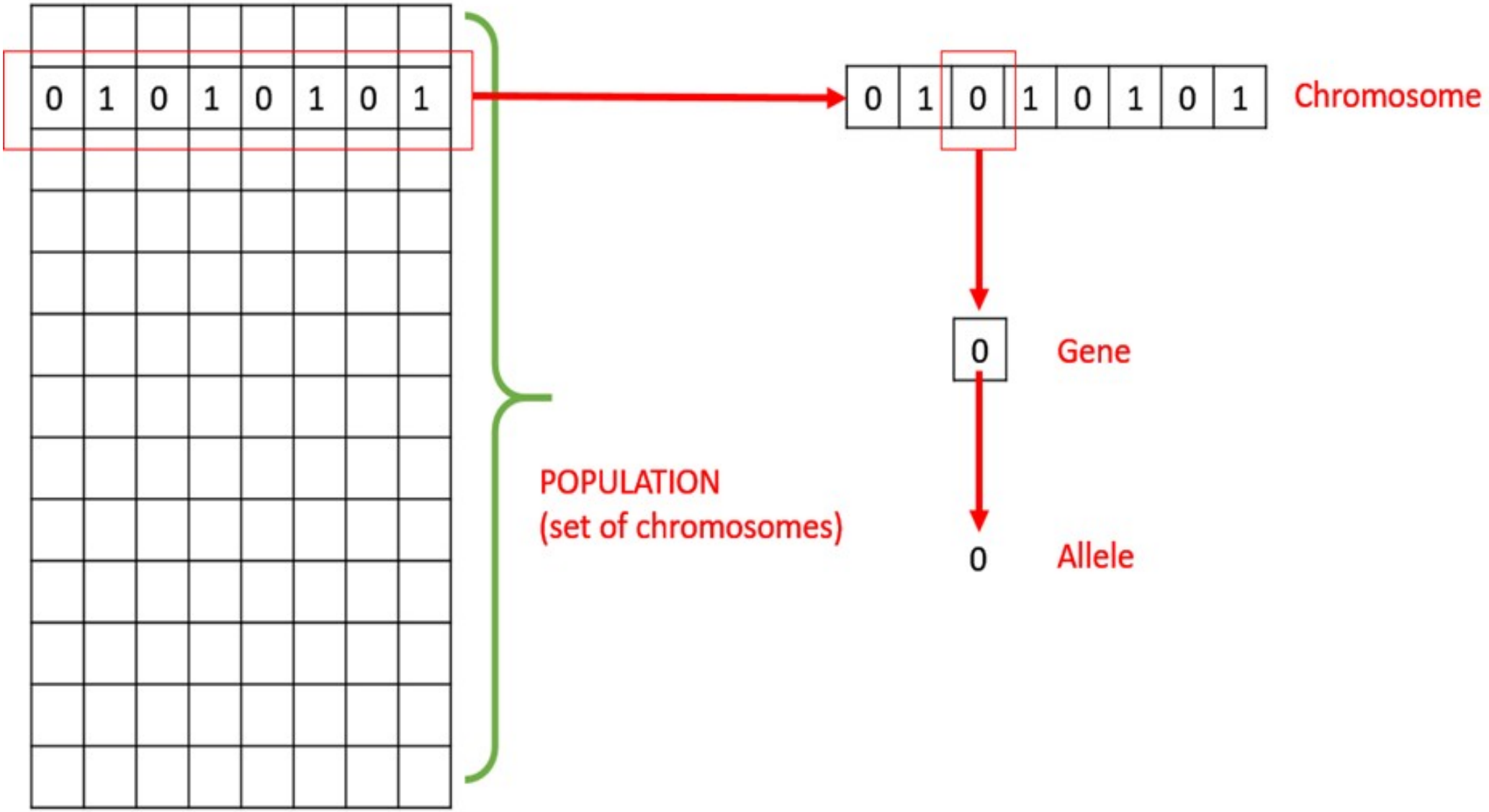
- Optimization is the process of making something better.

What are Genetic Algorithms?

- Nature has always been a great source of inspiration to all mankind. Genetic Algorithms (GAs) are search based algorithms based on the concepts of natural selection and genetics.
- GAs are a subset of a much larger branch of computation known as Evolutionary Computation.

BASIC TERMINOLOGY

- **Population** – It is a subset of all the possible (encoded) solutions to the given problem.
- **Chromosomes** – A chromosome is one such solution to the given problem.
- **Gene** – A gene is one element position of a chromosome.
- **Allele** – It is the value a gene takes for a particular chromosome.

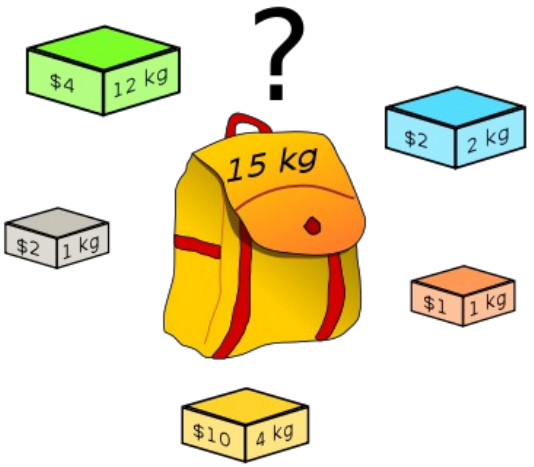

0 1 0 1 0 1 0 1
0 1 0 1 0 1 0 1
Chromosome
0
Gene
0
Allele
POPULATION
(set of chromosomes)

- **Genotype** – Genotype is the population in the computation space. In the computation space, the solutions are represented in a way which can be easily understood and manipulated using a computing system.
- **Phenotype** – Phenotype is the population in the actual real world solution space in which solutions are represented in a way they are represented in real world situations.
- **Decoding and Encoding** – For simple problems, the phenotype and genotype spaces are the same. However, in most of the cases, the phenotype and genotype spaces are different. Decoding is a process of transforming a solution from the genotype to the phenotype space, while encoding is a process of transforming from the phenotype to genotype space.Decoding should be fast as it is carried out repeatedly in a GA during the fitness value calculation.

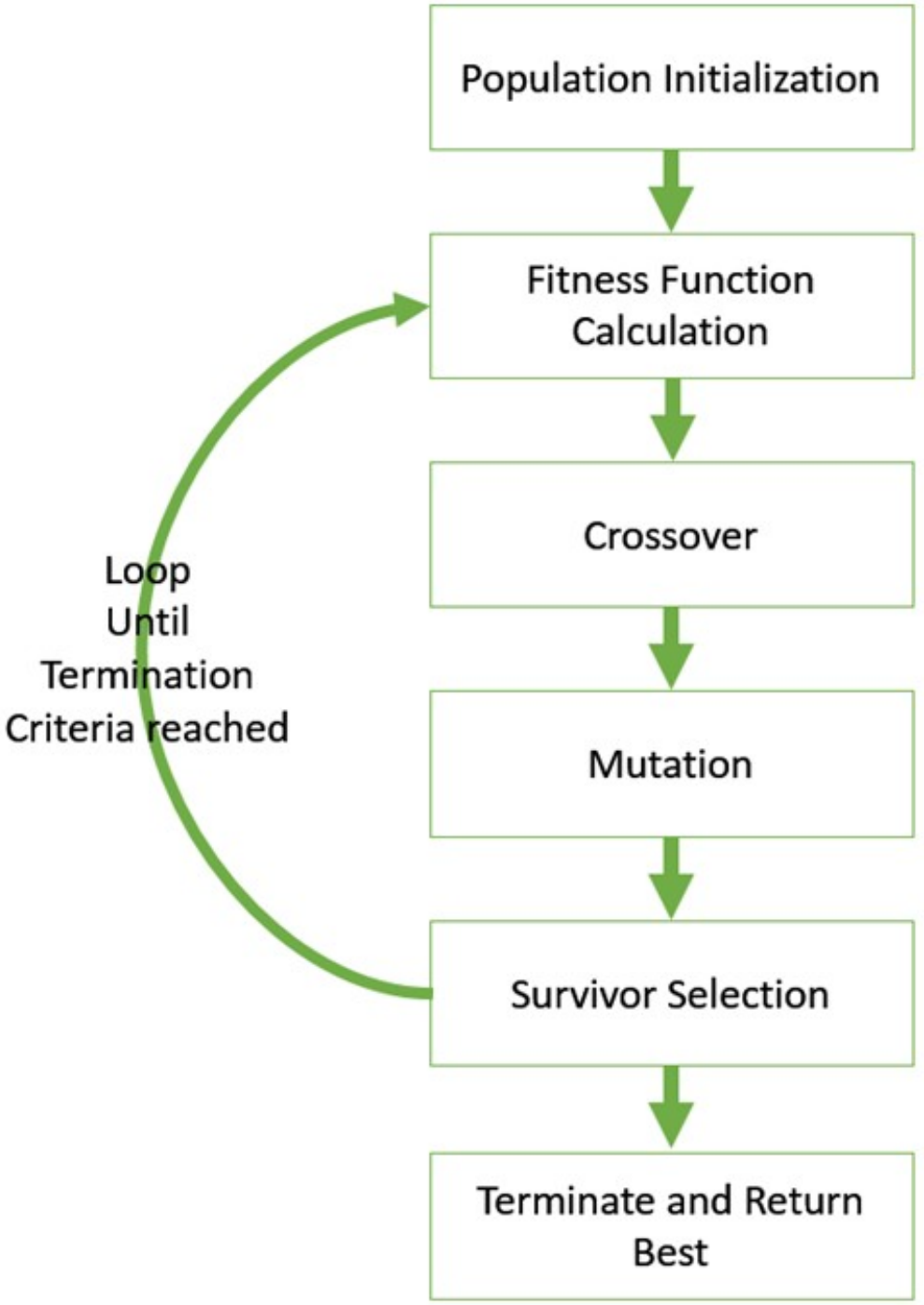
- **Fitness Function** – A fitness function simply defined is a function which takes the solution as input and produces the suitability of the solution as the output. In some cases, the fitness function and the objective function may be the same, while in others it might be different based on the problem.
- **Genetic Operators** – These alter the genetic composition of the offspring. These include crossover, mutation, selection, etc.

Knapsack Problem

- The knapsack problem or rucksack problem is a problem in combinatorial optimization: Given a set of items, each with a weight and a value, determine the number of each item to include in a collection so that the total weight is less than or equal to a given limit and the total value is as large as possible. It derives its name from the problem faced by someone who is constrained by a fixed-size knapsack and must fill it with the most valuable items.

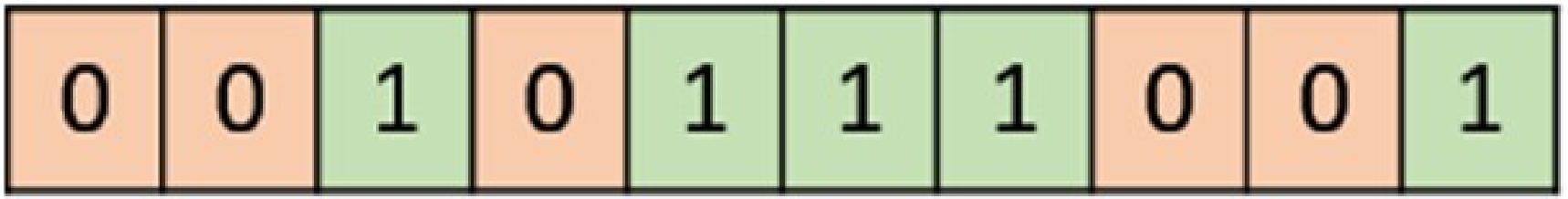


BASIC STRUCTURE OF GENETIC ALGORITHM

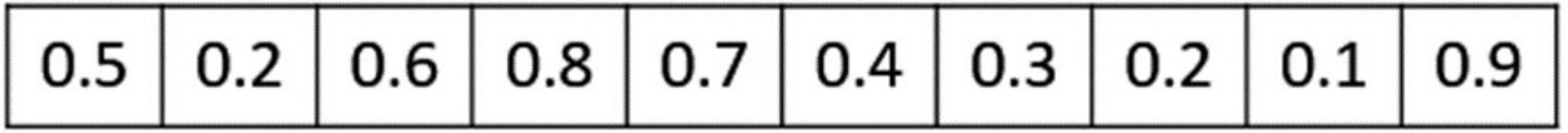


GENOTYPE REPRESENTATION

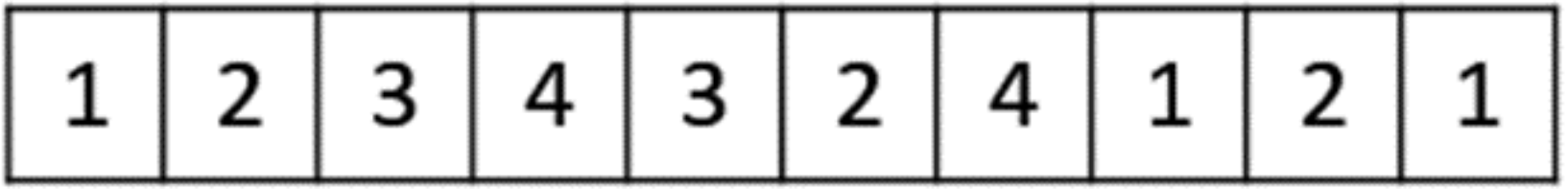
- BINARY REPRESENTATION



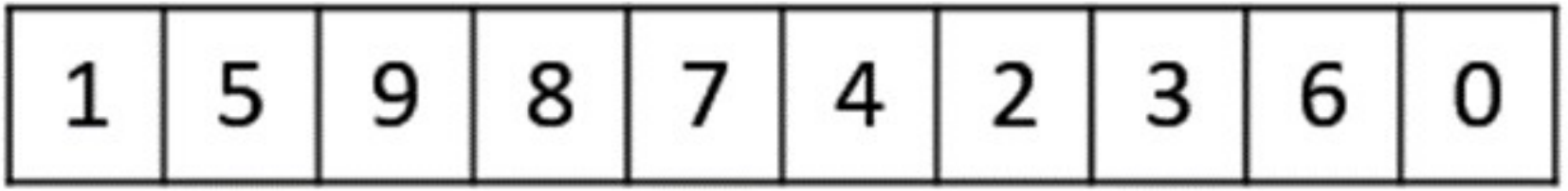
- REAL VALUED REPRESENTATION



- INTEGER REPRESENTATION



- PERMUTATION REPRESENTATION



GA- POPULATION

- Population is a subset of solutions in the current generation
- Set of chromosomes

- **POPULATION INITIALIZATION**
    1. RANDOM INITIALIZATION
    2. HEURISTIC INITIALIZATION

- **POPULATION MODEL**
    1. STEADY STATE
    2. GENERATIONAL

FITNESS FUNCTION

- Takes a candidate solution to the problem as input and produces as output
- The objective is to either maximize or minimize the given objective function

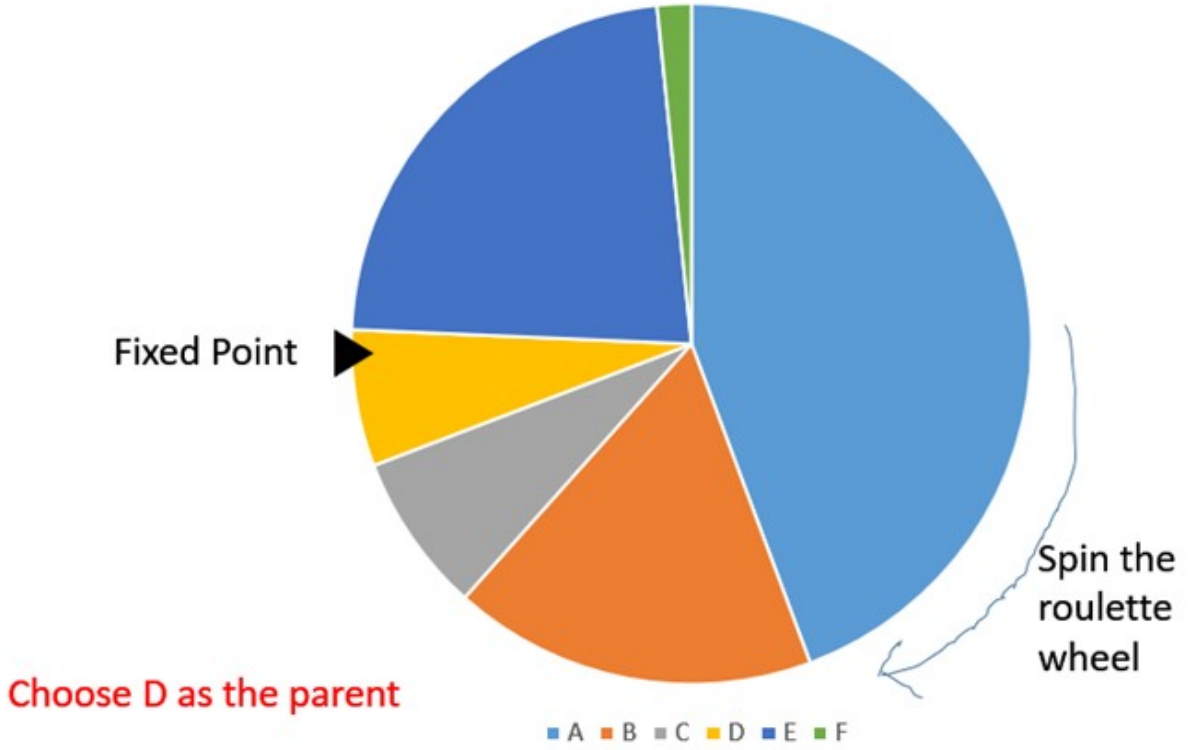
Knapsack capacity = 15

Total associated profit = 18

Last item not picked as it exceeds knapsack capacity

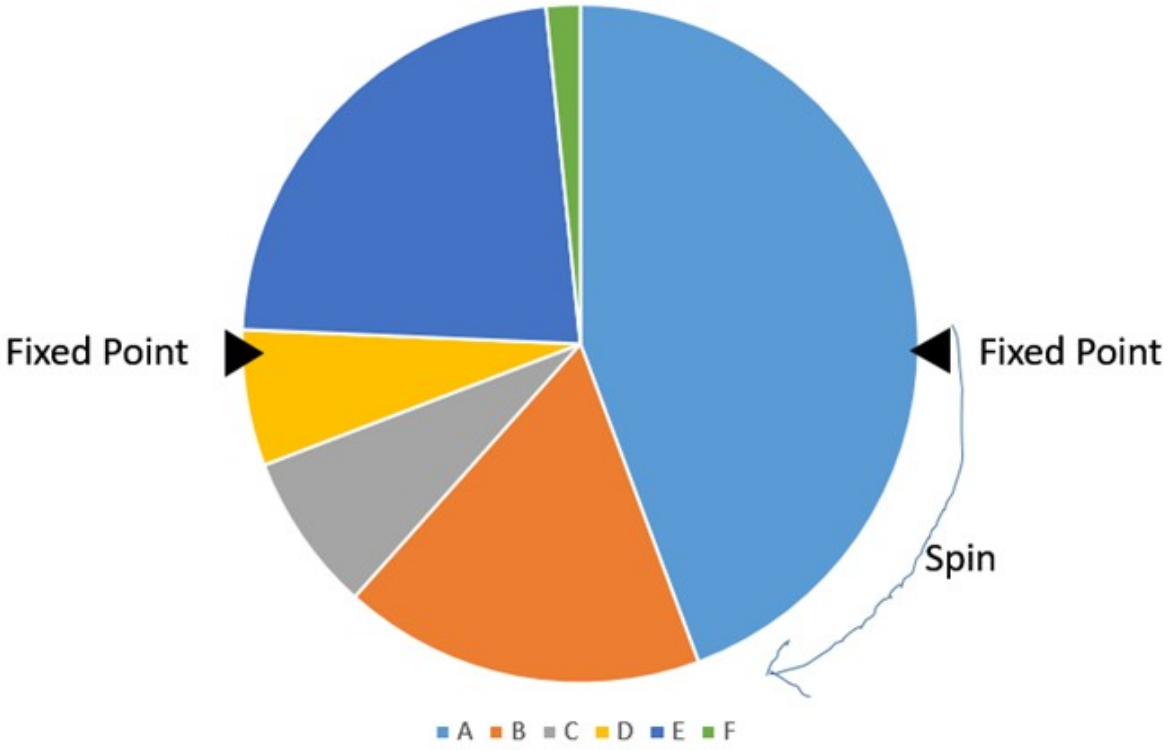
GA- PARENT SELECTION

- **Fitness Proportionate Selection**
  - **Roulette Wheel Selection**
    -



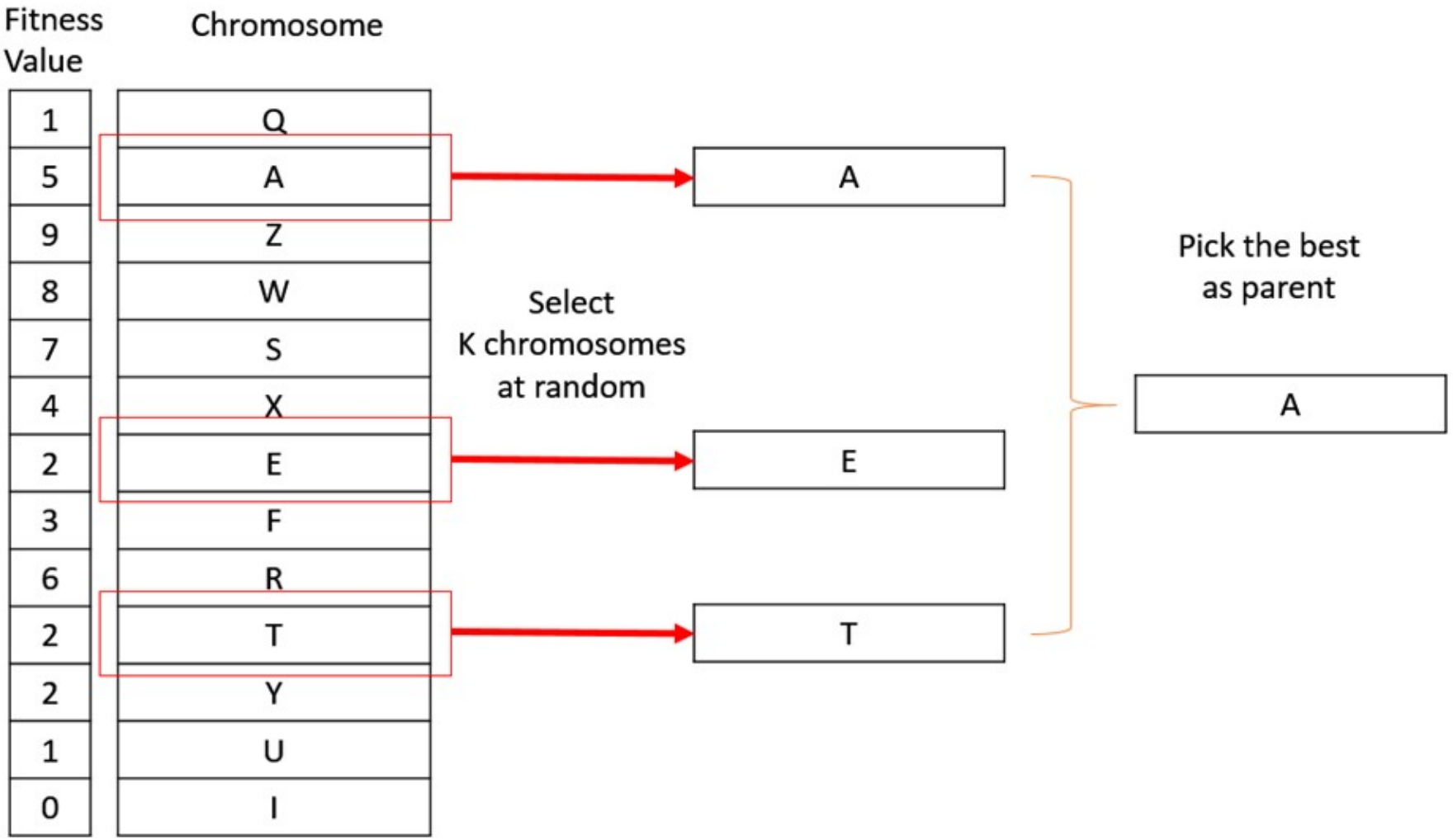
Chromosome	Fitness Value
A	8.2
B	3.2
C	1.4
D	1.2
E	4.2
F	0.3

- Stochastic Universal Sampling (SUS)



Chromosome	Fitness Value
A	8.2
B	3.2
C	1.4
D	1.2
E	4.2
F	0.3

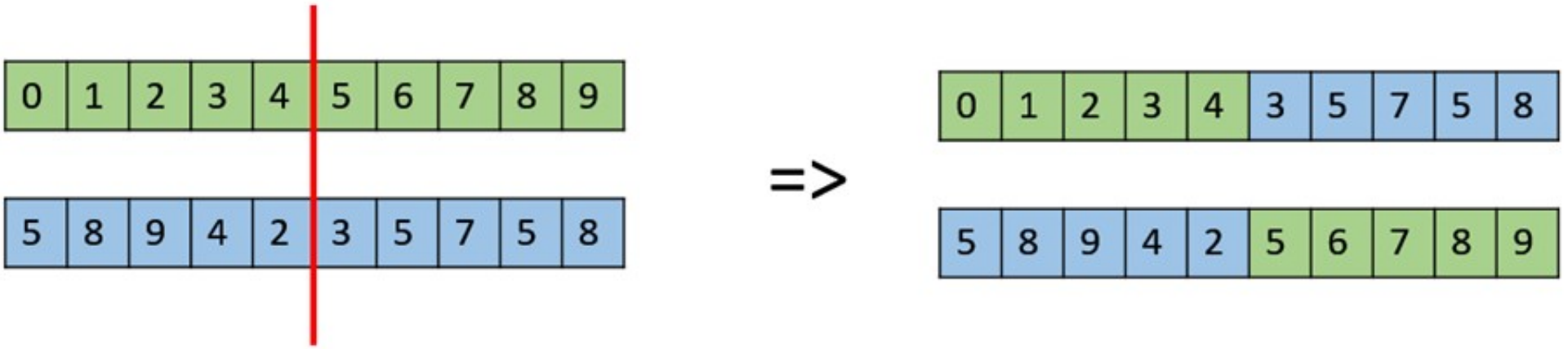
- Tournament Selection



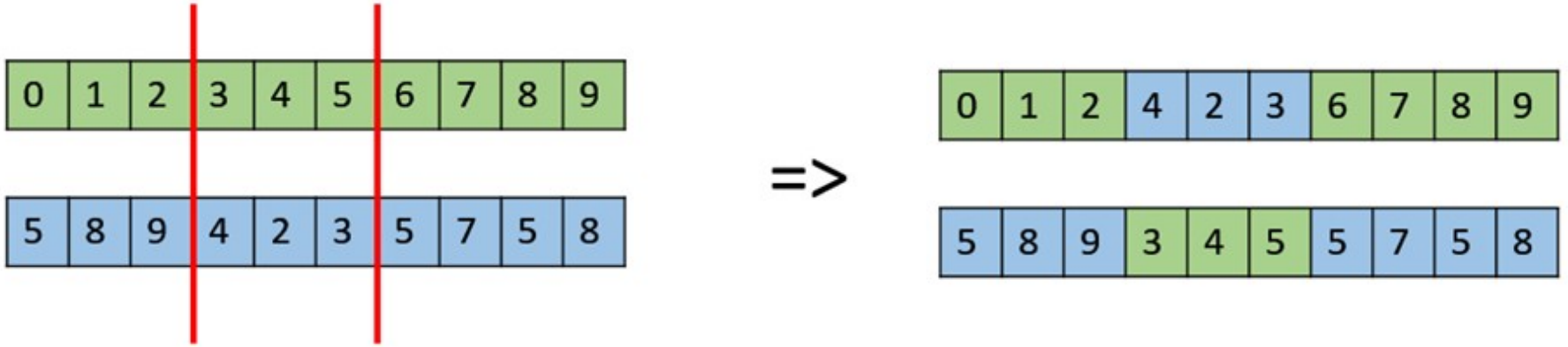
- Rank Selection

GA- CROSSOVER

- one parent is selected and one or more off-springs are produced using the genetic material of the parents
- **One Point Crossover**



- Multi Point Crossover

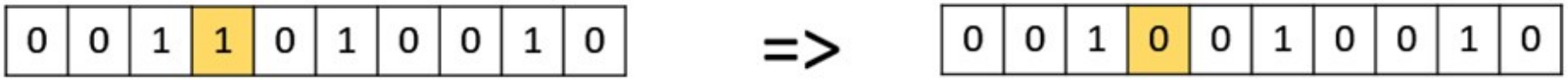


- Uniform Crossover

GA- MUTATION

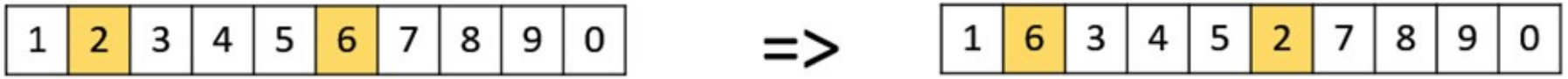
- used to maintain and introduce diversity in the genetic population
- Mutation Operators:

-Bit Flip Mutation

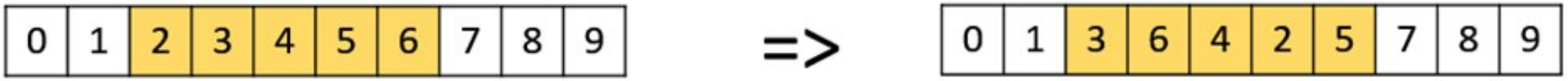


-Random Resetting

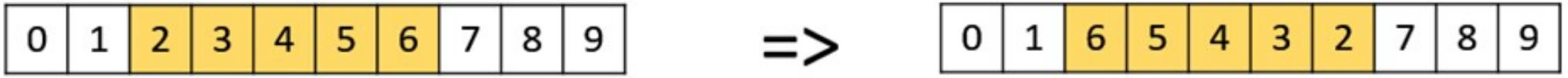
-Swap Mutation



- Scramble Mutation

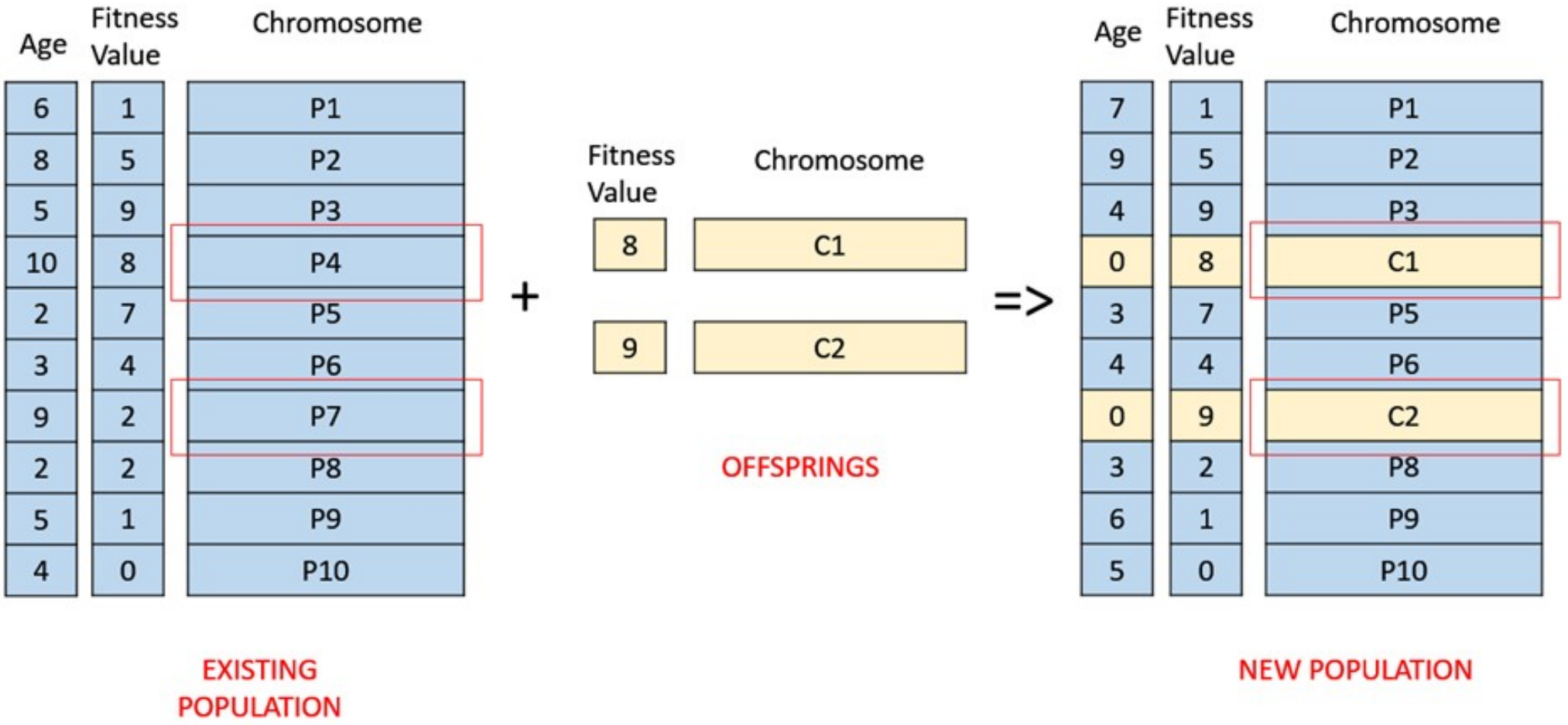


- Inversion Mutation

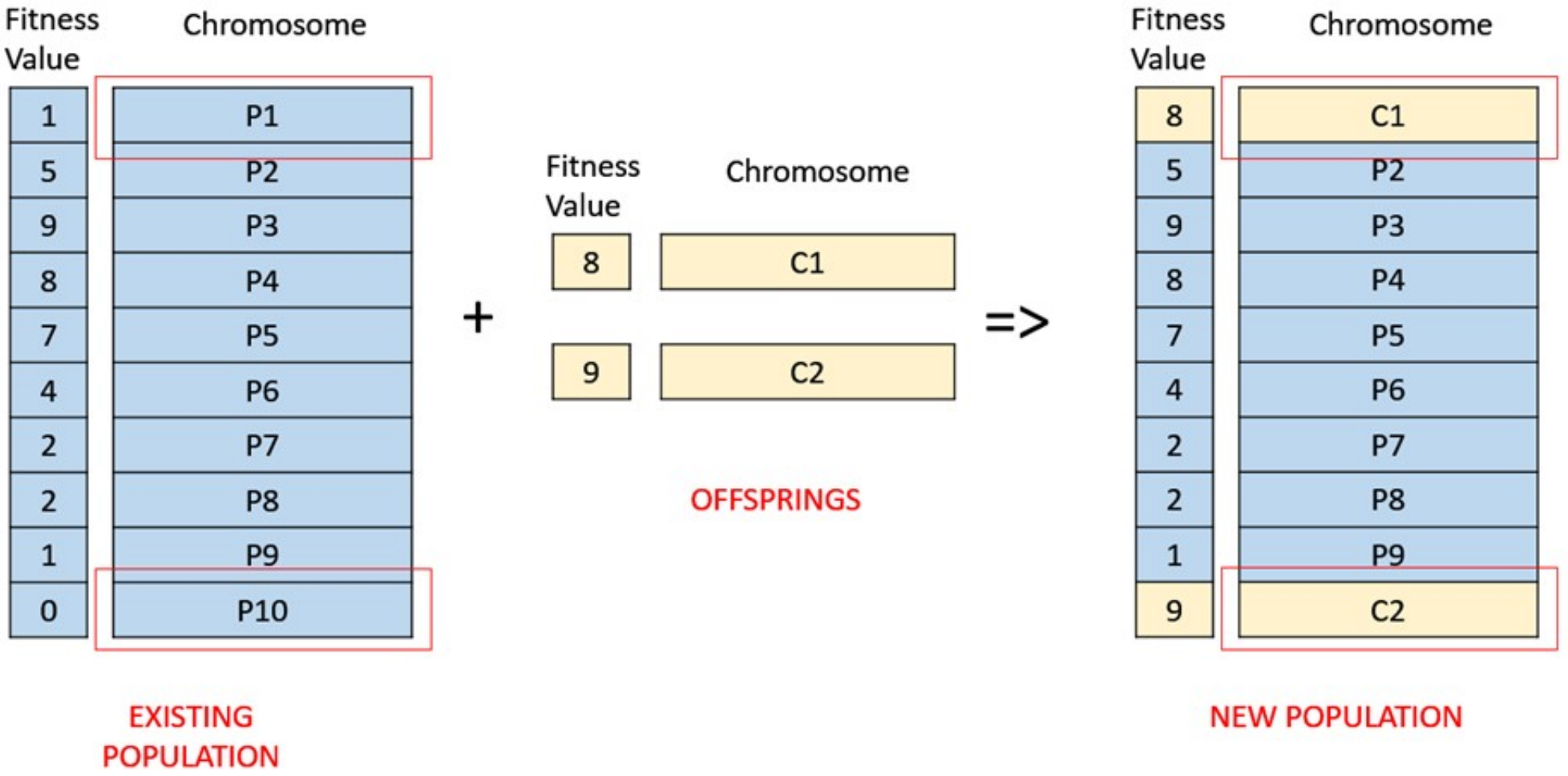


GA- SURVIVOR SELECTION

- Age Based Selection



- Fitness Based Selection



GA- TERMINATION CONDITION

- When there has been no improvement in the population for X iterations.
- When we reach an absolute number of generations.
- When the objective function value has reached a certain pre-defined value