**2. GENETIC ALGORITHM** (**GA**)

In [computer science](https://en.wikipedia.org/wiki/Computer_science) and [operations research](https://en.wikipedia.org/wiki/Operations_research), a **genetic algorithm** (**GA**) is a [met heuristic](https://en.wikipedia.org/wiki/Metaheuristic) inspired by the process of [natural selection](https://en.wikipedia.org/wiki/Natural_selection) that belongs to the larger class of [evolutionary algorithms](https://en.wikipedia.org/wiki/Evolutionary_algorithm) (EA). Genetic algorithms are commonly used to generate high-quality solutions to [optimization](https://en.wikipedia.org/wiki/Optimization_(mathematics)) and [search problems](https://en.wikipedia.org/wiki/Search_algorithm) by relying on bio-inspired operators such as [mutation](https://en.wikipedia.org/wiki/Mutation_(genetic_algorithm)), [crossover](https://en.wikipedia.org/wiki/Crossover_(genetic_algorithm)) and [selection](https://en.wikipedia.org/wiki/Selection_(genetic_algorithm)).

**Optimization problems**

In a genetic algorithm, a [population](https://en.wikipedia.org/wiki/Population) of [candidate solutions](https://en.wikipedia.org/wiki/Candidate_solution) (called individuals, creatures, or [phenotypes](https://en.wikipedia.org/wiki/Phenotype)) to an optimization problem is evolved toward better solutions. Each candidate solution has a set of properties (its [chromosomes](https://en.wikipedia.org/wiki/Chromosome) or [genotype](https://en.wikipedia.org/wiki/Genotype)) which can be mutated and altered; traditionally, solutions are represented in binary as strings of 0s and 1s, but other encodings are also possible.[[2]](https://en.wikipedia.org/wiki/Genetic_algorithm#cite_note-FOOTNOTEWhitley199466-2)

The evolution usually starts from a population of randomly generated individuals, and is an [iterative process](https://en.wikipedia.org/wiki/Iteration), with the population in each iteration called a generation. In each generation, the [fitness](https://en.wikipedia.org/wiki/Fitness_(biology)) of every individual in the population is evaluated; the fitness is usually the value of the [objective function](https://en.wikipedia.org/wiki/Objective_function) in the optimization problem being solved. The more fit individuals are [stochastically](https://en.wikipedia.org/wiki/Stochastics) selected from the current population, and each individual's genome is modified ([recombined](https://en.wikipedia.org/wiki/Crossover_(genetic_algorithm)) and possibly randomly mutated) to form a new generation. The new generation of candidate solutions is then used in the next iteration of the [algorithm](https://en.wikipedia.org/wiki/Algorithm). Commonly, the algorithm terminates when either a maximum number of generations has been produced, or a satisfactory fitness level has been reached for the population.

A typical genetic algorithm requires:

1. A [genetic representation](https://en.wikipedia.org/wiki/Genetic_representation) of the solution domain,
2. A [fitness function](https://en.wikipedia.org/wiki/Fitness_function) to evaluate the solution domain.

A standard representation of each candidate solution is as an [array of bits](https://en.wikipedia.org/wiki/Bit_array).[[2]](https://en.wikipedia.org/wiki/Genetic_algorithm#cite_note-FOOTNOTEWhitley199466-2) Arrays of other types and structures can be used in essentially the same way. The main property that makes these genetic representations convenient is that their parts are easily aligned due to their fixed size, which facilitates simple [crossover](https://en.wikipedia.org/wiki/Crossover_(genetic_algorithm)) operations. Variable length representations may also be used, but crossover implementation is more complex in this case. Tree-like representations are explored in [genetic programming](https://en.wikipedia.org/wiki/Genetic_programming) and graph-form representations are explored in [evolutionary programming](https://en.wikipedia.org/wiki/Evolutionary_programming); a mix of both linear chromosomes and trees is explored in [gene expression programming](https://en.wikipedia.org/wiki/Gene_expression_programming).

Once the genetic representation and the fitness function are defined, a GA proceeds to initialize a population of solutions and then to improve it through repetitive application of the mutation, crossover, inversion and selection operators.