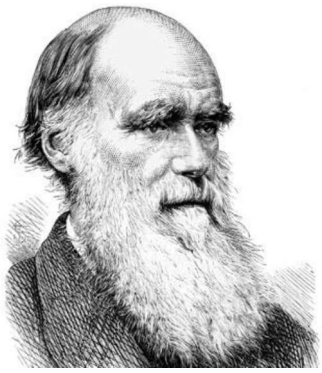
# Evolutionary Computing

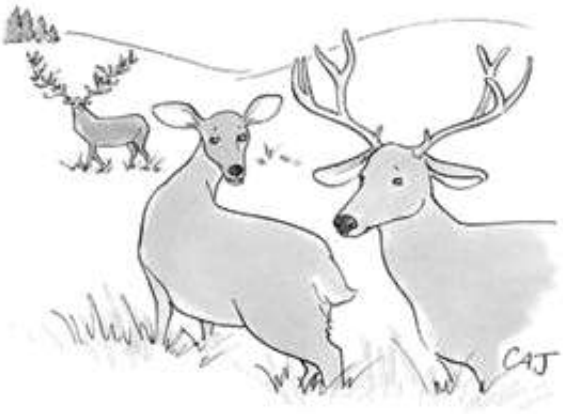
(Book: Eiben. Introduction to Evolutionary Computing)



Evolutionary Computing is a research area within computer science. As the name suggests, it is inspired by Darwin's theory of natural evolution.

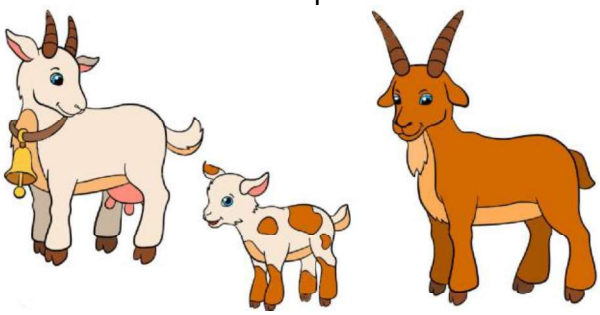
In an environment with limited resources, the fittest individuals survive. Also, they have more chances to have offspring.

Natural selection

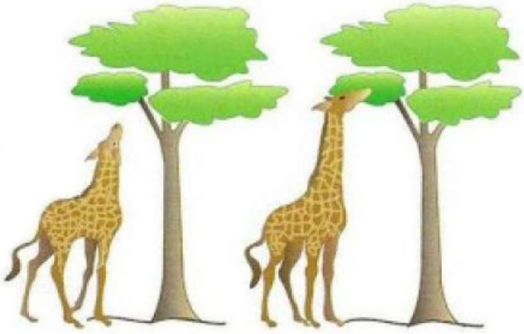


https://www.allposters.com/-sp/It-s-not-you-it-s-natural-selection-New-Yorker-Cartoon-Posters_i9184757_.html

Reproduction

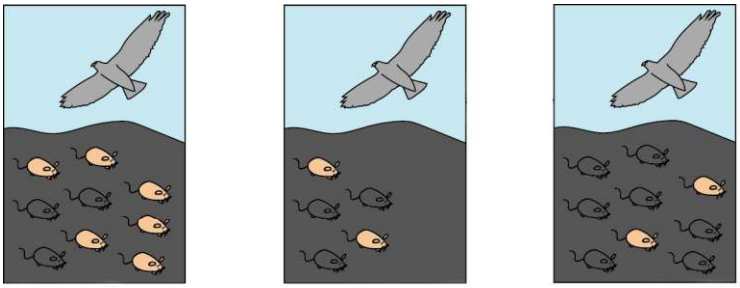


Mutation



http://www.mercados.lat/index.php/otros/item/1052-seleccion-natural

Survival of the fittest individuals



https://es.khanacademy.org/science/biology/her/evolution-and-natural-selection/a/darwin-evolution-natural-selection

| Evolution | | Problem solving |
|---|---|---|
| Environment | ↔ | Problem |
| Individual | ↔ | Solution |
| Fitness | ↔ | Quality |

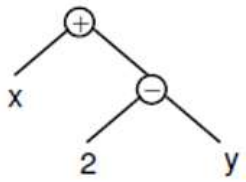
The main idea of all evolutive algorithms is the same: first, a ***population*** of ***individuals*** is randomly generated, and the ***fitness*** of all of them is calculated. Iteratively, where each iteration represents a generation, we do the following steps. The fittest individuals are ***selected*** as parents of the next generation, aiming to improve the average of the population's fitness. We use ***crossover*** for creating new individuals (offspring) based on parents. Some of the new individuals are ***mutated*** to explore new solutions. This process is executed until an individual is good enough or the computational limits have been reached.

```
BEGIN
  INITIALISE population with random candidate solutions;
  EVALUATE each candidate;
  REPEAT UNTIL ( TERMINATION CONDITION is satisfied ) DO
    1 SELECT parents;
    2 RECOMBINE pairs of parents;
    3 MUTATE the resulting offspring;
    4 EVALUATE new candidates;
    5 SELECT individuals for the next generation;
  OD
END
```

**Individuals' representation**: An individual represents a solution. The representation of the individual in the original problem is called ***phenotype***. The representation used for solving the problem is called ***genotype***. In other words, an individual's genotype encodes its phenotype. For example, the phenotype of the variable age is 4, but the genotype is 000100. Another example is:

Phenotype:
F(x,y) = x + (2-y)

Genotype:




**Fitness**: It is a numeric value that represents the quality of the solution.

**Population**: It is a group of individuals (solutions) that recombine and mutate their properties. The initial population is randomly created.

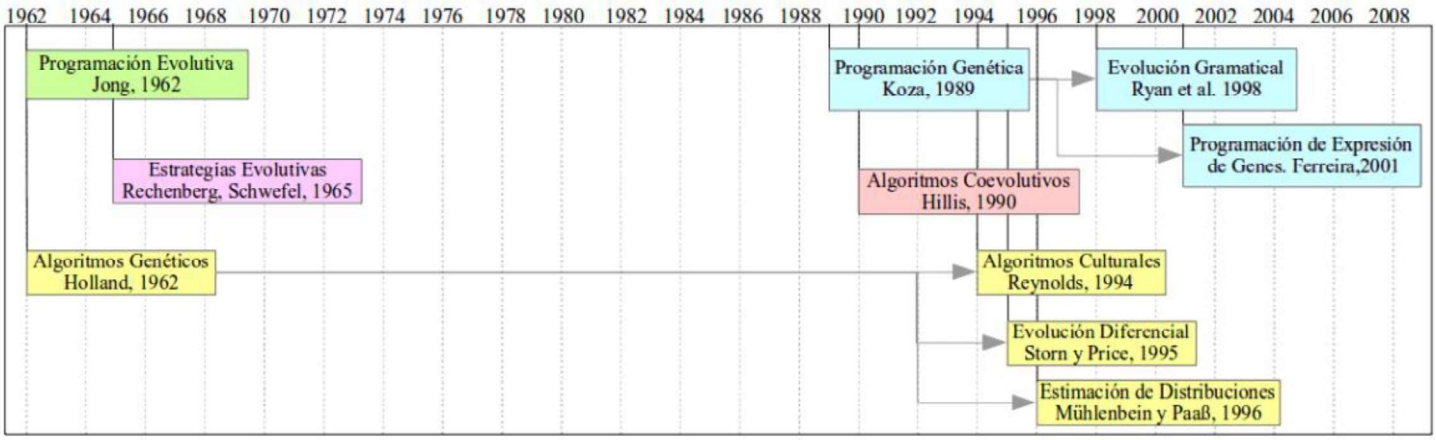
**Selection of parents**: For creating a new generation of individuals (offspring), the parents need to be selected based on their fitness. The main idea is to inherit the characteristics of the fittest individuals.

**Crossover (reproduction):** The parents inherit their characteristics to their offspring.

**Mutation**: Individuals modify their characteristics or behavior to improve themselves.

**Survival**: The fittest individuals survive and can live more time.

**Termination condition:** If you know the value of good fitness, the algorithm can stop when you find an individual with good fitness. However, generally, we do not know the values. In those cases, we can stop the algorithm when: the number of maximum fitness evaluations have been reached, after a defined number of iterations, or when the fitness of the best individual has not been changed after a defined number of generations.


1962 1964 1966 1968 1970 1972 1974 1976 1978 1980 1982 1984 1986 1988 1990 1992 1994 1996 1998 2000 2002 2004 2006 2008
Programación Evolutiva
Jong, 1962
Estrategias Evolutivas
Rechenberg, Schwefel, 1965
Algoritmos Genéticos
Holland, 1962
Programación Genética
Koza, 1989
Algoritmos Coevolutivos
Hillis, 1990
Evolución Gramatical
Ryan et al. 1998
Programación de Expresión
de Genes. Ferreira,2001
Algoritmos Culturales
Reynolds, 1994
Evolución Diferencial
Storn y Price, 1995
Estimación de Distribuciones
Mühlenbein y Paaß, 1996