

## Genetic Algorithm – The 8 Queens problem

### Table des matières

Genetic Algorithm – The 8 Queens problem.....	1
Introduction.....	1
Problem .....	1
Constraints .....	1
Optimum criteria .....	1
Fitness function and representation as a chromosome.....	1
Genetic operators.....	2
Mutation :.....	3
Simulation.....	3
Sources .....	3

### Introduction

This document is about the 8 Queens problem and its resolution with the Genetic Algorithm. In first the problem will be introduce. Then each steps of the genetic algorithm will be described, these step will be repeated until the optimum solution will be found.

### Problem

The 8 Queens problem consist in placing height queens in a chest board in such a place that there is no queen attacking each other.

### Constraints

As a constraint, we can only have one queen per row on the board.

### Optimum criteria

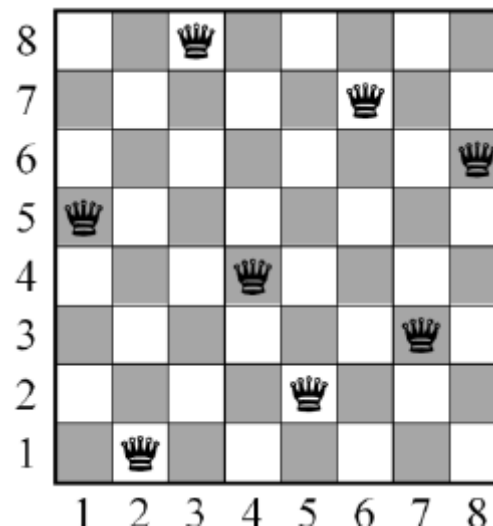
The maximum criteria is when one individual of the population get the maximum fitness value. His values will be the position of each queens on the board.

### Fitness function and representation as a chromosome

The fitness function is the number of pairs of nonattacking queens in a board, for each queen we count each other queens she doesn't attack.

To represent a board as a string we represent it as height positions. Each row is a position and for each row we tell at wich position is the queen. We begins by the left and tells the position of the queen with its position in y axis.

## GENETIC ALGORITHM – THE 8 QUEENS PROBLEM



Beginning by the left, the first queen of this board is in position 5 in row 1. With this logic we can represent the state as [51842736] and the fitness function equals to 28.

### Genetic operators

To demonstrate the genetic operators we'll took this initial population :

State	Fitness function
[24748552]	24
[32752411]	23
[24415124]	20
[32543213]	11

### Selection :

For the selection we randomly choose two pairs to reproduce based on the probabilities of each one. The chance of each individual to be selected is the percentage of its fitness function score by the addition of each fitness individual scores.

$$\text{Probability} = \text{fitness function} / \text{total fitness functions} * 100$$

State	Fitness function	Selection chance
[24748552]	24	31%
[32752411]	23	29%
[24415124]	20	26%
[32543213]	11	14%

For the example, the choosen individuals, based on their selection chance, will be :

State	Fitness function
[32752411]	23
[24748552]	24
[32752411]	23
[24415124]	20

### Crossover :

For each pair a crossover point will be choosed randomly, values will then be switched between individuals.

## GENETIC ALGORITHM – THE 8 QUEENS PROBLEM

Initial state	After crossover state
[327   52411]	[32748552]
[247   48552]	[24752411]
[32752   411]	[32752124]
[24415   124]	[24415411]

### Mutation :

To avoid to be stuck at a local optimal state, individuals need to have a chance to mutate.

After the crossover, for each values of an individual there will be a probability of 0.05 to mutate. If a mutation occurs, a random number between 1 and 8 will be selected to replace the current value.

### Simulation

For the simulation please see 8QueensProblem.py program compatible with Python 3.6.

The default values of the algorithm, the board size, the size of the population and the mutation will be asked at the beginning of the program.

The utility of the functions are describe at the top of them in the comments.

### Sources

Francisco Iacobelli « genetic algs » : [https://www.youtube.com/watch?v=qt85\\_CinKwo](https://www.youtube.com/watch?v=qt85_CinKwo)