



**Assignment.3 Selected topics in AI.2  
Genetic Algorithm (GA)**

	Name	ID
1	Esraa yazid Ahmed	20190093
2	Mariam Ihab Mohamed	20190801
3	Christina Montasser Saad	20190382

## **Table of contents :**

### **1. Methodology(How does GA work?)**

### **2. Evolve :**

- Evolve population
- The Selection Operator

### **3. Mutation :**

- Mutation Settings
- Mutation Strategy

### **4. Configuration and Results**

## 1. How does GA work ?

We have followed the known steps of genetic algorithm at the beginning we started by initializing our population with 20 chromosomes each chromosome contains (6 genes ) each of them is concerned with affecting one of the features that we have chosen to evaluate the chromosome as a whole(so we can say that genes of the chromosomes represent weights of used features) , here is a brief description to the features that we have used:

1. Scores : the default score calculated by the game configuration,, in our code we have initialized genes that will affect score variable with a weight within the range (1,10)
2. Bumpiness : bumpiness is a measure of how the height of the columns in tetris board differs ,high bumpiness will lead to difficulty in clearing rows in the game , in our code we have initialized genes that will affect bumpiness variable with a weight within the range (-6,-1)
3. Landing height :landing height is the height of the tallest column in the board on which i will place my peace , bigger of values of landing heights will lead the game to be harder as we will have more tall columns that will make it harder to place pieces between them to fill the gaps and holes , in our code we have initialized genes that will affect landing height variable with a weight within the range (-4,-1)
4. Clear rows after : this variable refers to the number of cleared rows that results from a falling piece that has fit in a gap and completed a horizontal line so the row is cleared , in our code we have initialized genes that will affect clear rows after variable with a weight within the range (1,4)
5. Number of holes : Number of holes can be defined as the number of cells (empty spaces ) that can not be accessed by other falling pieces because they are surrounded by other pieces from right , left and above it ; greater number of holes makes it hard to complete horizontal rows
6. in order to clear them , in our code we have initialized genes that will affect number of holes value with a weight within the range (-3,0)

7. Block transitions : Block refers to the number of perfectly connected pieces on the board, high number of blocks means more cleared rows. Block transitions, refers to the number of gaps between adjacent blocks, whether there is an edge or not we have calculated it vertically and horizontally in our code we have initialized genes that will affect block transitions value with a weight within the range (-4,-1)

---

## 2. Evolve :

- Evolve population

Evolve population is done using fitness function. We take the top 75% of the fittest individuals. Then, parents are selected based on the selection operator. For each pair of parents, the crossover and mutation are applied to the new child, and it is added to the new population. Also, to maintain a high level of fitness in the population over time and to ensure that the best solutions are not lost, 25% of the best individuals of the fittest individuals from the previous generation are added to the new population

- The Selection Operator

It is a simple tournament selection; It takes a list of best individual indices, which represent the indices of the fittest individuals in the population. Then, it randomly selects two individuals from that list. And finally, they are compared to each other using the fitness score, the individual with the higher fitness score is added to the list of candidates. It repeats the process until the required number of individuals are selected, and returns the list of candidate indices.

---

## 3. Mutation:

- Mutation Settings

The mutation rate is set to 0.02, which means that there is a 2% chance of a gene being mutated. For each gene in the chromosome, the mutation operator checks whether or not. This is done by generating a random number between 0 and 1, and comparing it to the mutation rate. If the random number is less than the mutation rate, a mutation will occur

- **Mutation Strategy**

It is a simple Gaussian mutation strategy, where each gene in the chromosome is mutated independently with a small random value, which is drawn from a uniform distribution. First, we generated a new gene value by selecting a random number between -2 and 2. This new gene value is added to the original gene value in the chromosome to create a new mutated gene.

---

## **4. Configuration and results :**

- **Seed :**

We have used a seed equal to 100

- **Results :**

The best two chromosomes are :

1. [[7.8375480729014715, -4.061839710999572, -2.0113816249913055, 2.380906310983931, -0.2406089390313637, -3.456002498876666]]  
And this chromosome resulted in the following score : [7928]
2. [5.621705573970637, -4.843370462068632, -3.8628706254586556, 1.8317335595358646, -0.2534419452340977, -3.456002498876666]  
and this chromosome resulted in the following score : [5884]