

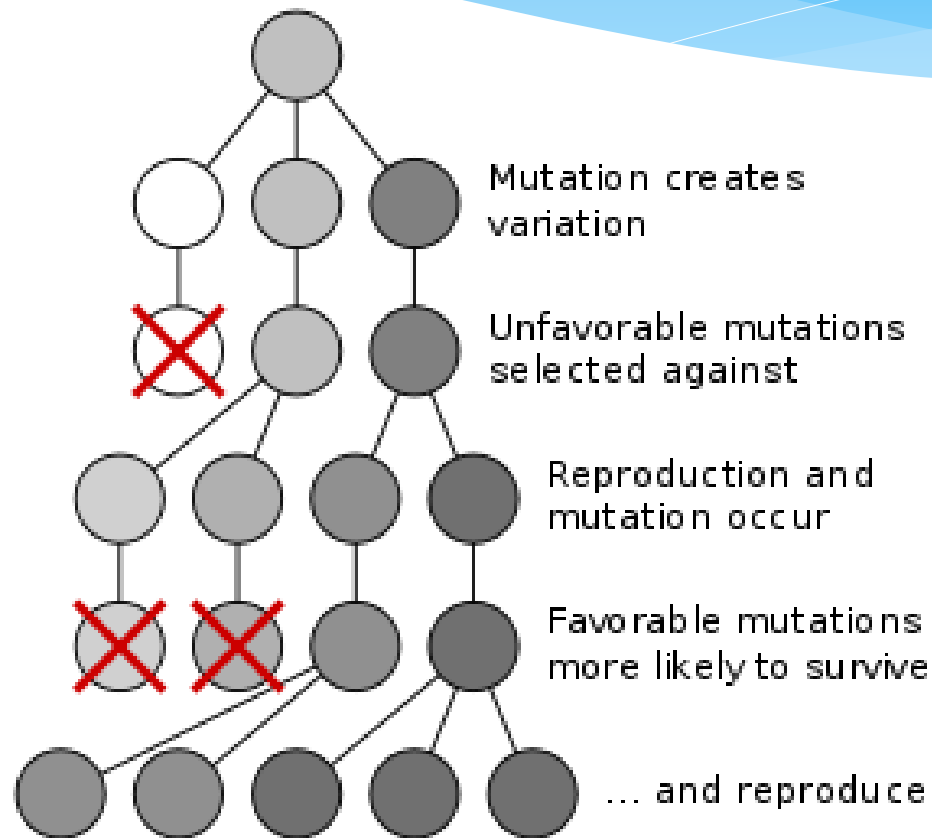
# Tetris and Genetic Algorithms

Math Club 5/30/2011

# What is a genetic algorithm?

- \* In biology, evolution is a process driven by random mutations, yet it produces very good results over time.
- \* A genetic algorithm is derived from nature's mechanism of evolution.

# How does evolution work?



# What are the mechanisms?

- \* Variation: Organisms in a population must be slightly different from one another.
- \* Inheritance: Traits of parent organisms must be passed onto their offspring.
- \* Limited space: Only some of the offspring in any generation is able to survive and pass on its genes.
- \* Competition: Individuals that are more *fit* are more likely to pass on their genes to the next generation.

# How do you write a genetic algorithm?

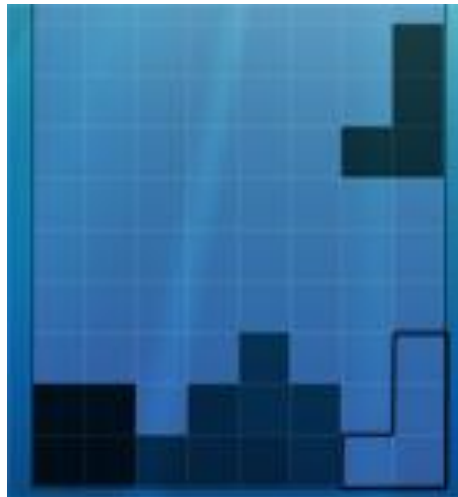
- \* A **chromosome** which expresses a possible solution to the problem as a string
- \* A **fitness function** which takes a chromosome as input and returns a higher value for better solutions
- \* A **population** which is just a set of many chromosomes
- \* A **selection** method which determines how parents are selected for breeding from the population
- \* A **crossover** operation which determines how parents combine to produce offspring
- \* A **mutation** operation which determines how random deviations manifest themselves

# A Tetris AI

- \* How do you make a computer program to play tetris?
- \* More generally, how do you play tetris?

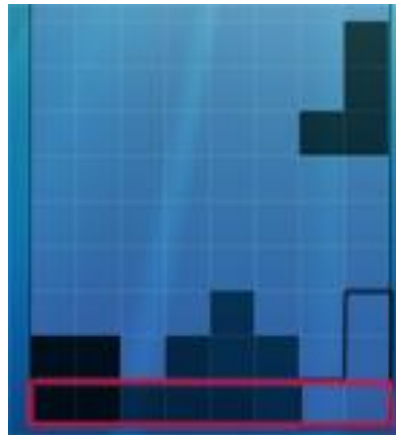
# Rules

- \* As I'm sure most of you know, in tetris you have blocks of four (*tetrominoes*) falling from the top of the board. The player moves and rotates the blocks and stacks them up:



# Rules

- \* Here the black outline is one of the places you can put the funny shaped block. And when a row is filled entirely with blocks (the row with the red outline below), you get a *clear*; that entire row is removed and the rest of the board is shifted down (often with a series of beeping noises and a small increase to your score):



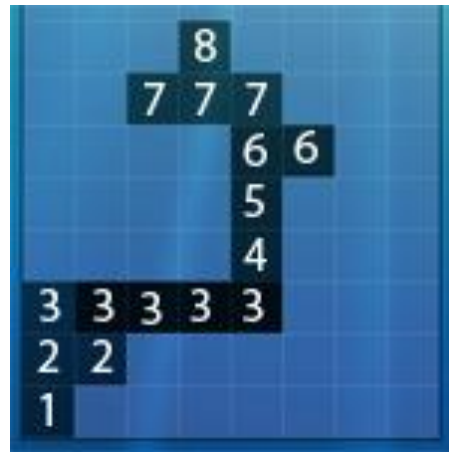


# Tetris AI - Outline

- \* We're going to make a method to score tetris positions (determine which positions are good and which are bad)
- \* The AI is going to go through each position and choose the best possible one.

# Strategies

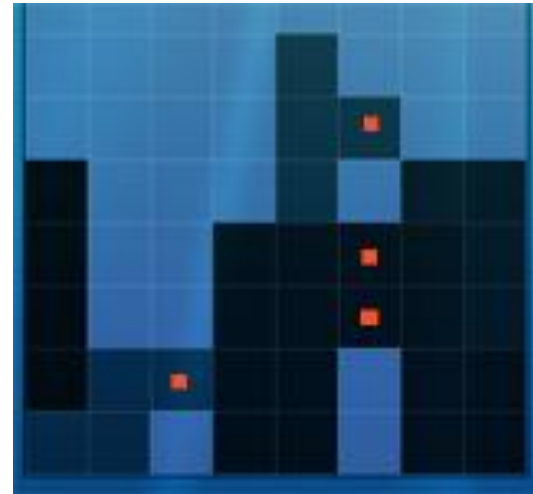
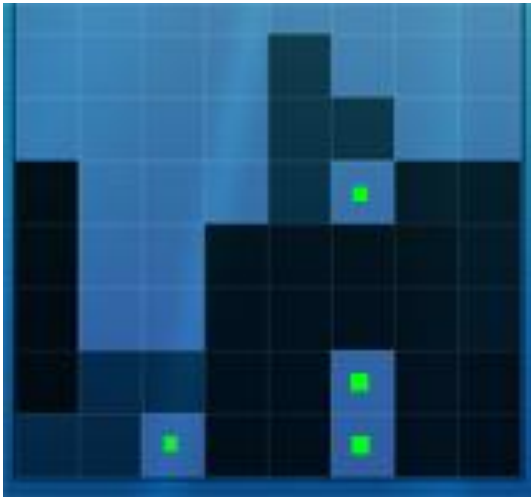
- \* It makes sense to penalize height since when all the blocks are stacked up to the top, you lose:



- \* It also makes sense to reward clears.

# Strategies

- \* It also makes sense to penalize holes and blockades:



# Putting together a tetris AI

- \* To get a score for a position, we would use something like this:
  - \* **Score = A \* Sum of Heights  
+ B \* Number of Clears  
+ C \* Number of Holes  
+ D \* Number of Blockades**
- \* Maybe A=-0.03, B=8.0, C=-7.5, D=-3.5?

# What about genetic algorithms?

- \* The problem is we are choosing the weights pretty much arbitrarily. They might work well or they might not, but we don't really know whether there are better values for them.
- \* Let's solve this problem by using a genetic algorithm to find the best weights for the AI.

# How do you write a genetic algorithm?

- \* A **chromosome** which expresses a possible solution to the problem as a string
- \* A **fitness function** which takes a chromosome as input and returns a higher value for better solutions
- \* A **population** which is just a set of many chromosomes
- \* A **selection** method which determines how parents are selected for breeding from the population
- \* A **crossover** operation which determines how parents combine to produce offspring
- \* A **mutation** operation which determines how random deviations manifest themselves

# Chromosome

- \* The chromosome is pretty simple – just an array of four weights.

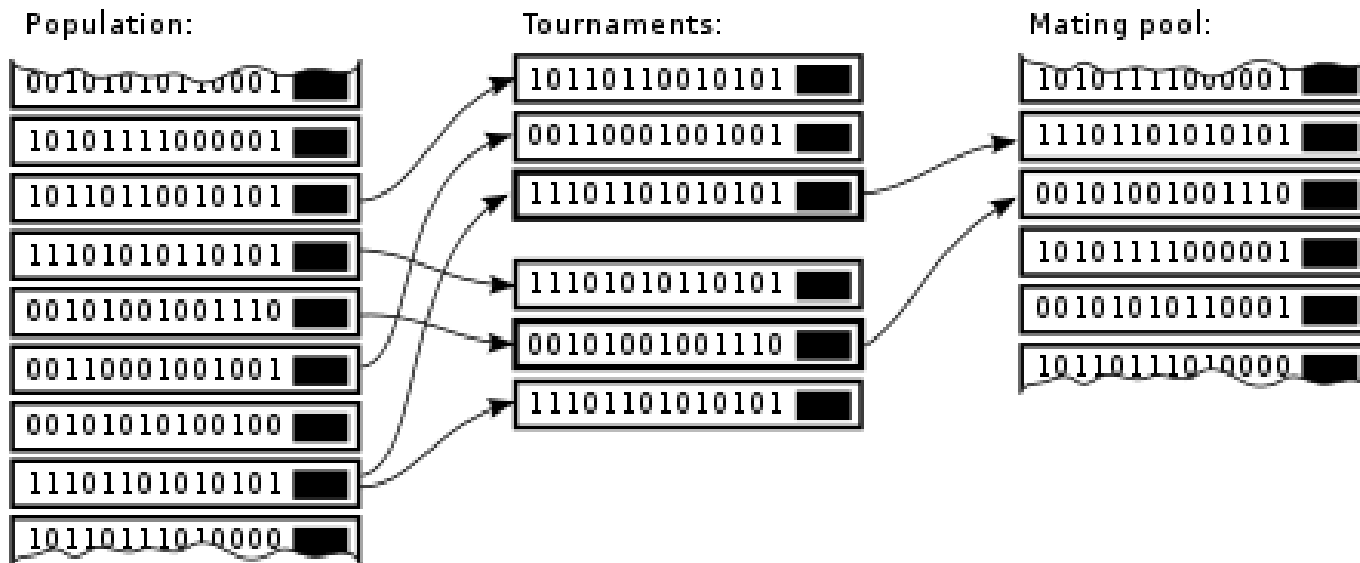
# Fitness Function

- \* The fitness function is also simple – the score is just the number of lines the AI runs for before it dies.



# Selection Method

- \* We are going to use a form of tournament selection:



# Crossover

- \* Also pretty simple, make the new chromosome by randomly choosing the four values from the parent chromosomes.

# Mutation

- \* We have a 10% chance of a mutation – a chromosome that is different from either parent's.

# The End

- \* Yes, we can play with the tetris now :D