# Players – players are selection algorithms

Players have 3 functions:

* Next Position – produces a position which it tries to insert into the current game
* Digest – consumes the result of adding a position to the game
  + Outcomes
    - Clear: Position was accepted in game
    - Row: Position was blocked by a queen on the same row
    - File: Position was blocked by a queen
    - Diagonal: Position blocked by a queen on a 45
    - Duplicate: Position already held by another queen
* Give up – returns a Boolean value if true the player has given up on the current game because they believe it’s a failed solution

## Rando Calrissian

Selects a row and a file position at random. This option was never coded as it would yield too much bad data. Plus calculating when to give up is a crap shoot.

## Once Burned

Maintains a list of possible file values and a list of possible row values. Keeps a guess limit initially equal to the N \* N.

* Next Position – randomly selects a row and a file from the remaining possible.
* Digest – consumes the result of adding a position to the game
  + Outcomes
    - Clear: Removes the file from the list of possible values and row from the list of possible rows. Resets guess limit to the size of row domain size multiplied by the file domain size.
    - Row: This is not a possible outcome
    - File: This is not a possible outcome
    - Diagonal: decrement guess limit
    - Duplicate: This is not a possible outcome
* Give up – returns true if the guess limit has reached 0

## Rolling Solution

The Rolling Solution simplifies the Once Burned Solution by observing that all solutions contain a queen on each row and file. It maintains a list of possible file values and increments a row value on success. It keeps a guess limit initially equal to N.

* Next Position – randomly selects a file from the remaining possible.
* Digest – consumes the result of adding a position to the game
  + Outcomes
    - Clear: Removes the file from the list of possible values. Increments the row. Resets guess limit to the size of row domain size.
    - Row: This is not a possible outcome
    - File: This is not a possible outcome
    - Diagonal: decrement guess limit
    - Duplicate: This is not a possible outcome
* Give up – returns true if the guess limit has reached 0

# Mutation – What can we mutate about the selection. What can w

# Fitness Factors

## Duplication – We should punish fitness for duplicating existing games

## Failure – We should punish fitness for making more games that are noise

### Degree – Failures can be ranked by how close to a solution they got. If n=8 then a game with 7 placements is better than a game with 4.

### Rate

## Satisfaction