# CSE 443 – PATTERN RECOGNITION HOMEWORK 2 REPORT

In this homework,  I write a program that write a snake playing program. For this, subfield of genetic algorithms where every inividual of  the population is a simple program represented by a tree of commands are used.

Program tree is made up of terminals(leaves) and functions(internal nodes). In this case 3 terminal and 5 internal nodes are used.

**Terminals**

**move_forward :** the snake would maintain its current direction, and move forward.
**turn left :** the snake would change its current direction, making move to the left.
**turn right :**  the snake would change its current direction, making move to right.

**Internal Nodes(Functions)**

**ifFoodAhead:** if there is food in line with the snake's current direction, this function will execute its first arugment, otherwise it will execute second argument. This was the only initial function that gave the snake information beyond its immediate surrondings.

**ifDangerAhead:** if the game square immediately in front of the snake is occupied with either a snake body segment or the wall, this function will first execute its first argument, otherwise it will execute its secon argument.

**İfDangerLeft:** if the game square is immediately to the left of the snake occupied with either snake body segment or the wall, this function will execute its first argument, otherwise it will execute its first argument, otherwise it will execute its second argument.

**İfDangerRight:** if the game square is immediately to the right of the snake occupied with either snake body segment or the wall, this function will execute its first argument, otherwise it will execute its first argument, otherwise it will execute its second argument.

**Progn2:** This is connectivity function that will first execute its right argument, then its left. İt is the only function that allows execution of more than one terminal in a single parse of the function tree.

**Fitness:**  The fitness measure used pieces of food eaten.

**Parameters:** Population was set to 1000. The maximum number of generations was set to 500. The maximum depth of function tree was set to 10.

**Crossover:**  Subtree exchanges are used as crossover.

**Mutation:** Subtree replacements are used as mutation operator.