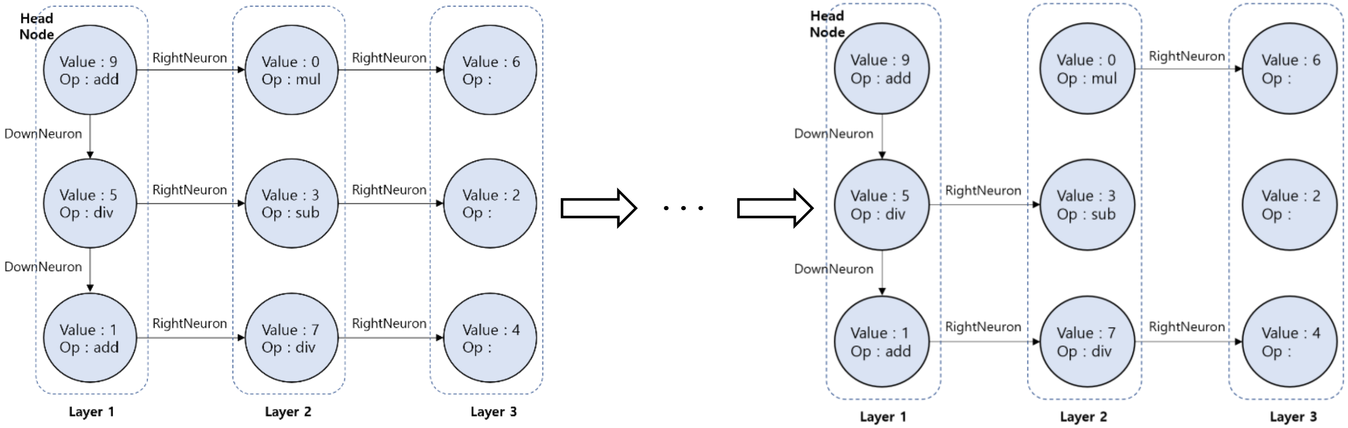
Object Oriented Programming

assignment 3-3

ASSIGNMENT 3-3

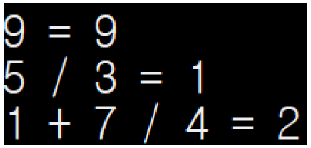
Overview

- There are 100 billion neurons in the brain. Among many neurons responsible for arithmetic computation, the connection of particular nine neurons can be disconnected according to the certain rules. Implement a program that simulates the operation of nine neurons according to the several rules.
  - Nine neurons initially construct a linked list as below
  - When state updated, each inter-layer connection can be broken
  - At final state, execute the calculation and print the result



<Initial state>

<Final state>



```
9 = 9
5 / 3 = 1
1 + 7 / 4 = 2
```

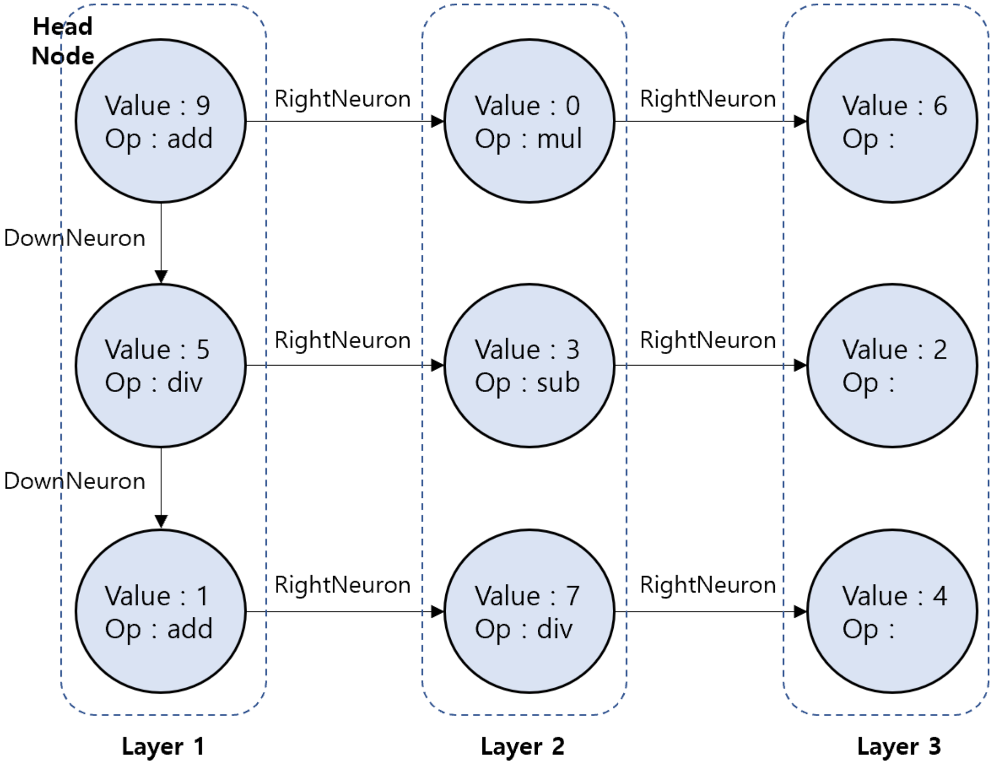
Class Neuron & Linked list

- **Class Neuron**
  - Member variables (You can use another data type)
    - int **Value** : Integer value N, 0<=N<=9
    - Char*/String **Op** : One of four operators of addition(add), subtraction(sub), multiplication(mul), and division(div)
    - Neuron* **RightNeuron** : Address of forward Neuron object connected with current neuron among the neurons of the next layer.
    - Neuron* **DownNeuron** : Address of forward Neuron object connected with current neuron among the neurons of the same layer.

- **Linked list** consists of nine Neuron objects.

First state

- Nine Neuron objects created and linked list is constructed as shown in the figure below.
- Member variables 'Value' and 'Op' is set as figure below.

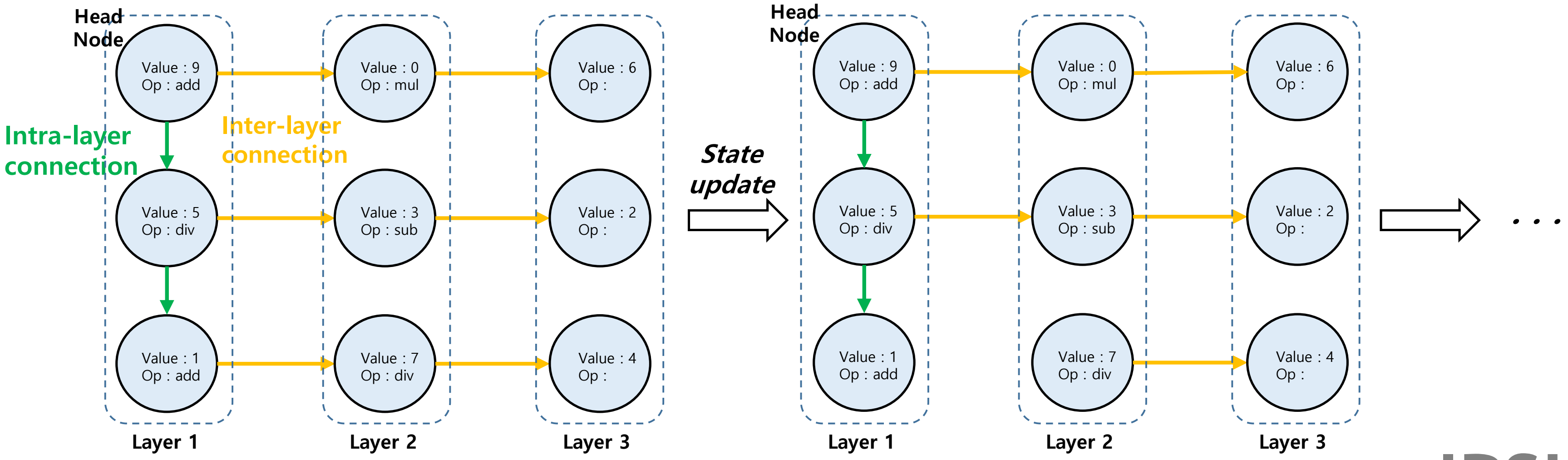


State update

- When updating the state, each inter-layer connection is broken with a 10% probability
- Total three state updates are performed in order
  - First state ->Second state, Second state->Third state, Third state->Fourth state
- The connection break is determined as shown in the following code

```
srand(1);      // Write only once at program start
ConnBreak = rand()%10;     //connection will be broken if value of break is zero
```

- Seed of rand operation should be given with srand(1) for deciding connection will be broken or not

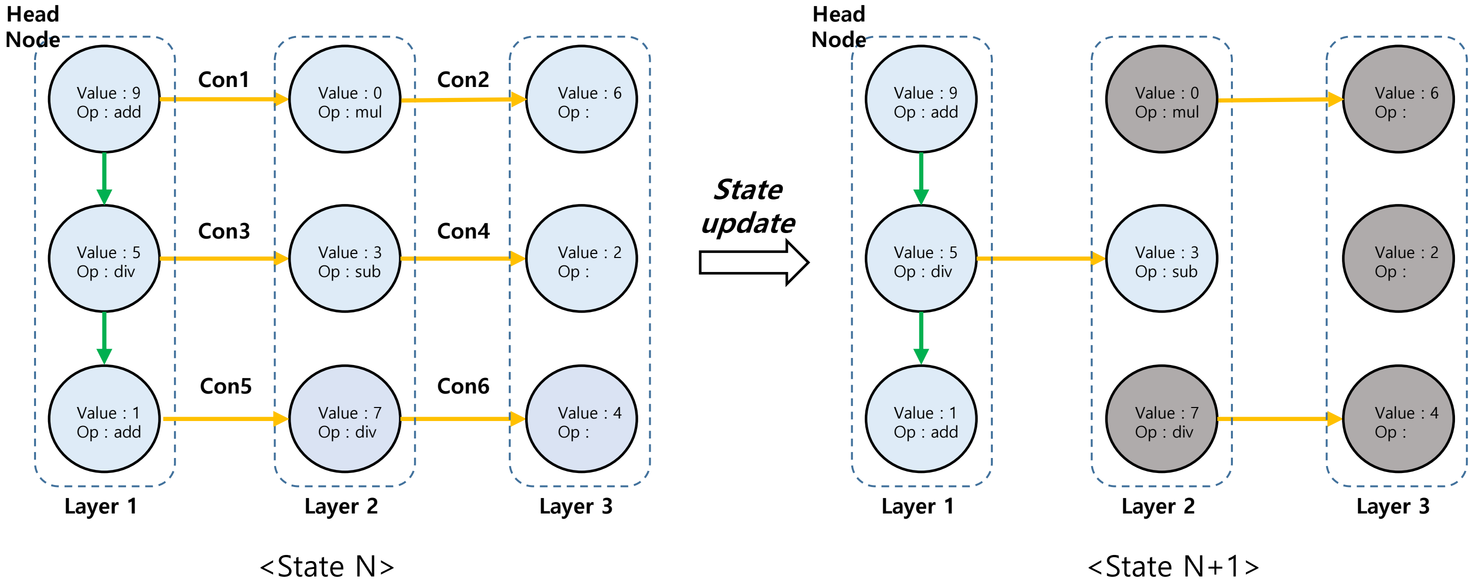


<First state>

<Second state>

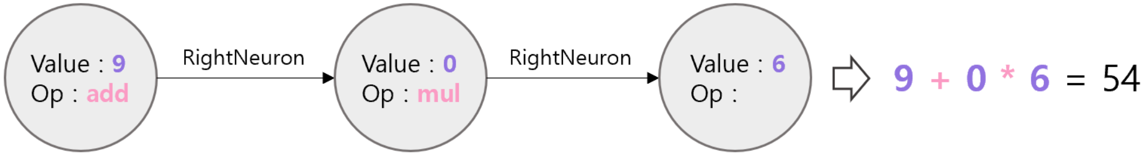
State update

- When updating the state, only the connections of the nodes accessible from the head node are considered in ascending order of connection index(Con1->Con2->Con3->..)
  - In the example below, state update from State N to State N+1 proceeds as below
    - Con1 is disconnected -> Do not consider disconnection of Con2
    - Con3 is connected -> Consider disconnection of Con4
    - Con5 is disconnected -> Do not consider disconnection of Con6
- Node not accessible from the head node, gray color nodes in the example below, may be deleted at any time

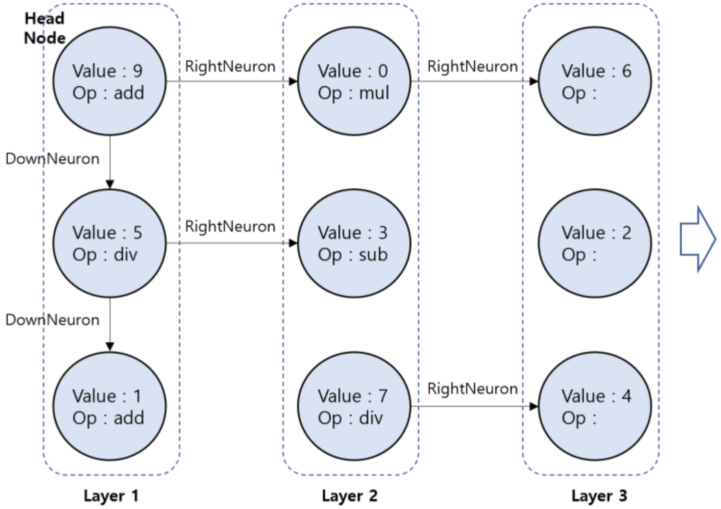


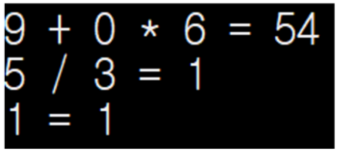
Final state

- In the fourth state, calculation will be executed horizontally from each neuron of first layer until there is no connection
- The horizontal calculation between the layers is proceeds sequentially from the previous layer



- Print the expression and answer
- Do not consider the priority of operators
- Discard the remaining of division
- If there is no connection between first layer and second layer, only the number of first neuron can be displayed or right term can be displayed also





```
9 + 0 * 6 = 54
5 / 3 = 1
1 = 1
```

<Final state>