Ryan Erickson

00783840

HW8

1.

a. java.lang.Math

b. class

c. yes

d. yes, java.lang.StrictMath

e. public int volts(double t1, double t2, float v1){

float s = (name of synapse).strength;

float v2 = v1 \* exp(t1-t2) + s;

return v2;

}

2.

a. synapse fires at time t. schedule its destination neuron to be kicked at time t.

b. synapse fires at time t. scheudle its destination neuron to be kicked at time t.

and neuron is kicked at time t. adjust the votage on that neuron, and if the result is that the neuron should fire,

schedule this neuron to fire at time t.

3..a.

1.Neuron fires at time t. schedule each outgoing synapse to be kicked at time t.

2.c

3.synapse fires at time t. scheudle its destination neuron or destination synapse to be kicked at time t.

4.if nueron is kicked at time t, adjust voltage on that neuron and if the result is that the neuron should fire,

schedule this neuron to fire at time t or if it is a synaspe, increment voltage by the strength of the synapse.

go to the first step to check if it is eligible of a neuron fire

b.

1.Neuron fires at time t. schedule each outgoing synapse to be kicked at time t.

2.synapse is kicked at time t. schedule this synapse to fire at time t+delay

3.synapse fires at time t. scheudle its destination neuron to be kicked at time t. (added bullet point) - or the destination

may be a primary synapse, increase the voltage by the strength of the destination synapse. schedule its destination again.

4. nueron is kicked at time t, adjust voltage on that neuron and if the result is that the neuron should fire,

schedule this neuron to fire at time t