Pseudo Code for Service Node Simulation







// Arrival times and service time are specified and given

StaticDeterministicSSQSimulation

1 

2 

3 **while** 

4 **if** 

5 

6 

7 

8 **if** 

9 

10 

11 

12 **else**

13 **if** 

14 ;

15 

16 **if** 

17 

18 **else**

19 

20 

21 

22 **end**



**//** Arrival times and service times are to be stochastically specified in advance  
// Arrival times, service times, and completion times are separately determined.

StaticStochasticSSQSimulationA

1 

2 

3 

4 **for****to** *n*-1

5 

6 

7 StaticDeterministicSSQSimulation 

8 **end**

// Arrival times, service times, and completion times are determined in one-pass.

StaticStochasticSSQSimulationB

1 ;

2 **for****to** *n*-1

3 

4 

5 

6 

7 StaticDeterministicSSQSimulation

8 **end**

// Stochastic arrival and service times are dynamically generated.

DynamicStochasticSSQSimulation

1 ;

2 

3 

4 **while**

5 **if** 

6 

7 

8 

9 **if** 

10 

11 **else**

12 

13 

14 

15 

16 **if** 

17 

18 

19 **else**

20 

21 **else**

22 

23 

24 

25 

26 

27 **if** 

28 

29 

30 

31 

32 **else**

33 

34 

35 

36 **end**

// Stochastic arrival and service times are dynamically generated, as well as the immediate feed back case is stochastically determined.



DynamicStochasticSSQWithFeedbackSimulation

1 ;

2 

3 

4 **while**

5 **if**  //arrival

6 

7 

8 

9

10 if x > 0 // server busy

11 q ← q + 1; Queue\_Plot( , q )

12 else // server free

13 Δ ← Uniform(,);

14 ← + Δ

15 x ← x + 1; Sevice\_Plot( , x)

16 τ ←

17 if count < n

18 Δ ← Exponential(μ)

19 ← + Δ; count ← count + 1

20 else

20.5 ← ∞

21 **else** //complete

22 

23 

24 if feedback not occured

25 l ← l -1; Node\_Plot( , l )

26 x ← x-1; Sevice\_Plot( , x)

27τ ←

28

29 if q > 0

30 q ← q – 1; Queue\_Plot( , q )

31 x ← x + 1; Sevice\_Plot(, x)

32 Δ ← Uniform();

33 ← + Δ

34 

35 

36 **end**



DynamicStochasticMultiServerSingleQueueSimulation

1 ;

2 

3 

4 **while**

5 **if**  //arrival

6 t ← ;

6.5 

7 repeat i

8 if = 0

9 free\_sever\_index = i

10 if free\_sever\_index != null

11 = 1

12 Service\_Plot( , )

13 Δ ← Uniform(,)

14 ← + Δ

15

16

17 else // all server busy

18 q ← q + 1; Queue\_Plot( , q )

19

20 τ ←

21

22 if count < n

23 Δ ← Exponential(μ)

24 ← + Δ; count ← count + 1

25 else

26 ← ∞

27 **else** //complete

28 

29 

30

31 l ← l -1; Node\_Plot( , l )

32 x ← x-1; Sevice\_Plot( , )

33τ ←

34

35 if q > 0

36 q ← q – 1; Queue\_Plot( , q )

37 x ← x + 1; Service\_Plot( , )

38 Δ ← Uniform();

39 ← + Δ