COMPUTER SCIENCE AND ENGINEERING

Indian Institute of Technology, Palakkad

CS3210: Computer Networks Lab

Lab 8 (FIFO queuing and Token bucket)

30 Sep, 2019

Time: 3 hrs

IIT PALAKKAD

1. Write a program called "**shape**" that implements token bucket based traffic shaping. This function should have two input arguments. The first argument is an integer and denotes the size of token bucket, whereas the second argument is the rate at which tokens are falling into the token bucket. The file "arrivals.txt" contains the list of packets arrivals. Each packet arrival is denoted by a line in this file as "**X Y Z**", where **X** is arrival time, **Y** is packet ID, and **Z** is packet length. You should be able invoke your program as follows:

   ./shape 250 4.0 < arrivals.txt

   The above invocation should shape the traffic in the file "arrivals.txt" using a token bucket of capacity 250 and rate 4.0. The output of the above invocation should be, for each packet, "**U V W**" (without quotes), where **U** is the transmission time of the packet ("%.2f" format), **V** is the packet ID ("%d" format), and **W** is length of the packet ("%d" format). Your program will be evaluated for 6 test cases by running the script "./testShape", and 5 point will be awarded for each test case passed. [30]

2. Write a program called "**fifo**" that implements a finite capacity FIFO queue. This function should have two input arguments. The first argument is an integer and denotes the buffer size (including the packet being served / HOL packet), whereas the second argument is the constant rate at which data is pushed to the output line of the queue. The HOL packet is held in buffer until it is completely transmitted. The file "arrivals.txt" contains the list of packets arrivals. Each packet arrival is denoted by a line in this file as "**X Y Z**", where **X** is arrival time, **Y** is packet ID, and **Z** is packet length. You should be able invoke your program as follows:

   ./fifo 250 4.0 < arrivals.txt

   The above invocation should simulate the traffic in the file "arrivals.txt" passing through a FIFO queue of capacity 250 and output rate 4.0. The output of the above invocation should be, for each packet, "**U V W**" (without quotes), where **U** is the transmission completion time of the packet ("%.2f" format), **V** is the packet ID ("%d" format), and **W** is length of the packet ("%d" format). Your program will be evaluated for 6 test cases by running the script "./testFifo", and 5 point will be awarded for each test case passed. [30]

3. The traffic in file "arrivals.txt" is first passed though a token bucket of capacity 500 and rate x, and then passed through a FIFO queue of capacity 1000 and rate 10.0. Find the largest x (upto 6 decimal places) which ensures that no packets are dropped by the FIFO queue? [40]