Computer Networks Lab Report

Md Talha Yaseen Khan 111701019

Lab-6 (Round-robin and Weighted-fair queuing)

- ### Common Function Overview
  - Firstly I defined two structure, **struct Packet** and **struct Queue**.
  - Packet structure contains arrival time of packets, packet ID, queue ID, length of packets and a pointer of next packet.
  - Queue structure contains pointer of two packets (head and tail) Where head points to first index of queue and tail to last index.
  - Function **int max(int a, int b)** returns maximum of a and b.
  - Function **bool isEmpty(Queue *queue)** tells queue is either empty or not.
  - Function **void enQueue(Queue *queue, Packet *packet)** used to push packets in queue.
  - Function **void deQueue(Queue *queue)** used to pop packets from queue.
  - Function Function **Packet *top(Queue *queue)** returns top packet of queue.

1. Round-robin Queuing

- ### Code Overview
  - Function **void rr(double rate)** simulates packet into round-robin way. Argument of this function is the output rate. We have 4 different queues here. So it is looking at queue in order 1 -> 2 -> 3 -> 4 -> 1 ... . When any queue is found empty it is choosing the next higher index queue in same order from the previously transmitted packets queue. It is printing the output transmission time alongwith packet ID of each packets.

- **Command to compile and run are:**

  gcc rr.c -o rr

  ./testRoundRobin.sh

2. Weighted-fair Queuing

Code Overview

- Function **void wfq(double rate)** simulates the weighted fair queue. Argument of this function is the output rate. Each time it transmitting a packet using function transmission().
- Function **bool transmission(Queue *q, double rate)** enqueuing packets in order of arrival time. And it is calculating the virtual finish time for each of the head packets of the queue. Virtual finish time $(F_i = max\{A_i, F_{i-1}\} + L_i/W_i)$. This function returns bool value, **true** if all queue are empty otherwise **false.** It is printing the output transmission time alongwith packet ID and queue ID of each packets.
- **Command to compile and run are:**

  gcc wfq.c -o wfq

  ./wfq 4.0 1.0 1.0 1.0 1.0 < arrivals.txt

- *Output of above is in the file out_wfq.txt attached in the zipped folder.*