

Name: Muhammad Fahad

ID: FA19-BSSE-0014

Course: Operating Systems

Section: AM

Teacher: Osama khan

CS 3220-AM, BM, CM, DM: Operating Systems

Assignment 2

Date Assigned: Nov 23rd, 2021

Q 1. In a real-world application of an online Airline Ticketing Reservation System that processes all customers' requests, provide names and quantities of all possible threads that might be needed in a process, where a process is associated with a single user using the system. (18 points)

S#	Thread	Quantity
1	User Identification	30
2	Account Selection	30
3	Balance Inquiry	15
4	Cash Withdrawal	15
5	Bill Payment	5
6	Money Transfer	5
7	Money Deposit	3
8	PIN Change	2

Q 2. In this problem you are to compare reading a file using a single-threaded server and a multithreaded server. It takes 30 msec to get a request for work, dispatch it, and do the rest of the necessary processing, assuming that the data needed are in the block cache. If a disk operation is needed, as is the case half of the time, an additional 50 msec is required, during which time the thread sleeps. How many requests/sec can the server handle?

- a. If it is single-threaded? (8 points)
- b. If it is multi-threaded? (8 points)

Solution:

Total time to read a file from Cache = $T_c = 30msec = 30 \times 10^{-3} \text{ second}$

Total time to read a file from Hard Disk = $T_D = 50msec + 30msec = 80 \times 10^{-3} \text{ second}$

a. If it is single-threaded? (8 points):

Average time to read a file (get a request for work, dispatch it, and do the rest of the necessary processing)

$$\begin{aligned} &= (T_c + T_D) / 2 \\ &= (30 + 80) / 2 \\ &= 110 / 2 \\ &= 55msec \\ &= 55 \times 10^{-3} \text{ second} \end{aligned}$$

Total number of requests per second that the server can handle = $1 / 55 \times 10^{-3} \text{ seconds} = 18.18 \approx 18$

b. If it is single-threaded? (8 points):

Average time to read a file (get a request for work, dispatch it, and do the rest of the necessary processing)

$$\begin{aligned} &= \min (T_c + T_D) \\ &= \min (30 + 80) \\ &= 30msec \\ &= 30 \times 10^{-3} \text{ second} \end{aligned}$$

(Average time to read a file would be the time it takes to read a file from Cache only.)

Total number of requests per second that the server can handle = $1 / 30 \times 10^{-3} \text{ seconds} = 33.33 \approx 33$