Matthew Huynh

CS350

January 31, 2011

HW 0

1) a. 10 hours for 1k/hour, 5 hours for 2k/hour => 20,000 hits / 15 hours = 1333.33 hits/hour

b. Take each element in the set of N observations and figure out how many hours the observation actually lasted. Find the set’s total hits and divide it by the set’s true hour total to get the mean.

2) a. 1 packet = 12,000 bits => 1,000,000,000 bits sec-1 / 12,000 bits = 83,333 packets/sec

b. 7500 / 83333 = 0.09

c. 0.9 – 0.09 = x / 83333 => x = 67500 packets => inject at least 67500 packets

3) a. The capacity is 100 because if you follow the path from A to B choosing the maximum capacity edge at each fork, the smallest capacity edge is 100. This means that this edge is the bottleneck and any traffic that uses this edge will be limited to 100.

b. I would upgrade the top edge of 10. Even through the edge immediately following that would bottleneck traffic to 100, the top edge would now be able to hold more traffic than 10 (similar to how a highway can have more lanes).

4) a. You can calculate the theoretical speedup using Amdahl’s Law:

1 / ((1 – P) + P/S) => 1 / ((1 – 0.1) + 0.1/2) = 1.05 times faster (or 5% faster)

b. I would choose the subsystem that is utilized the most and so it is the bottleneck. In this case, it is I/O. Using Amdahl’s Law again:

1 / ((1 – 0.625) + 0.625/2) = 1.4545 (or 45% faster)

5) a. 3/ 8 = 0.375

b. 5/8 = 0.625

c. 6/8 = 0.75 utilization, which requires 2 processes

d. If we increase the number of processes beyond this maximum, the processes in the queue would starve (depending on the scheduling algorithm, which ones would starve would be different).

e. If we increased the I/O’s speed to 3ms, the overall time would be 6ms. The CPU and I/O split would be 50% so 2 processes could run. One process would run on CPU while the other ran on I/O, then they’d swap, and repeat. The CPU utilization would be 100%.

6) a. You can take the average of the three utilization numbers because the subsystems are independent. If all 3 systems are at 1.0 utilization each, the total is 3.0 and the average is 1.0.

b. I would recommend speeding up the backend server because a web transaction spends the most time in there (50%).

c. You can calculate the speedup using Amdahl’s Law:

1 / ((1 – 0.5) + 0.5/2) = 1.33 times faster (33% faster)