Answer Set 1

3 FEBRUARY, 1994

Problem 1

In the hands on mode, I will assume that the user fills the entire 15 minute slot. Only 4 jobs can be processed an hour, and the processor works 1 useful minute out of every 15:

$$T = 4, \quad \eta = 1/15 = 7\%.$$

Given a job sequencing monitor, the sequence of events is a tireless repetition of:

... *read process print read process print* ...

Each read is 300 cards at 1000 cards/min, each print is 500 lines at 1000 lines/min, and processing is one minute. Hence a job takes 1.8 minutes. Therefore:

$$T = 60/1.8 = 33.\bar{3}, \quad \eta = 1/1.8 = 55.5\%$$

Problem 2

A tape leaves the central processor and:

1. travels for 5 minutes to the input-output computer
2. spends 25 minutes printing:

   $$50 \text{ jobs } \times 500 \text{ lines/job } \times .001 \text{ minutes/line}$$

3. spends 15 minutes reading:

   $$50 \text{ jobs } \times 300 \text{ cards/job } \times .001 \text{ minutes/card}$$

4. travels for 5 minutes back to the central processor

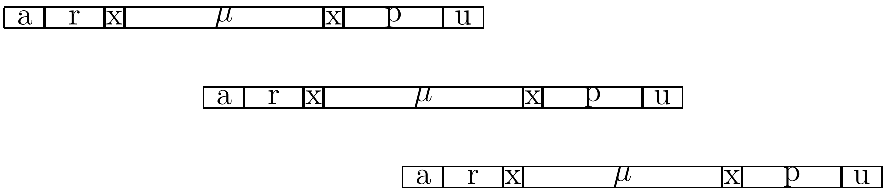
arriving 50 minutes after it left. This leaves just enough time for a second tape, loaded with 50 jobs, to be processed by the central processor, at 1 minute a job. Since the processor is always occupied:

$$T = 60, \quad \eta = 100\%.$$

The previous paragraph described how 100% processor utilization occurs by mounting 1 tape (with 50) jobs every 50 minutes on the processing computer. Hence the 10 minute bursting and batching activities occur at 50 minute intervals. Reading to printing takes 100 minutes, plus 20 minutes for bursting and batching. However, a job arrives, on average, in the middle of the period between batchings. Hence average waiting time is:

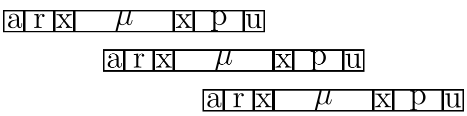
$$T = 100 + 20 + 25 = 145 \text{ minutes}.$$

Is such a schedule attainable? We draw a picture of the activities and verify that reading and printing are not required to be simultaneous:

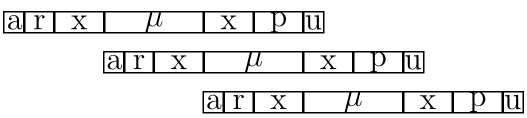


where a is batch, r is read, x is transport, μ is process, p is print and u is burst.

We will try to reduce T by using smaller job batches. Suppose batching jobs by 25 takes 5 min. to batch and burst, 5 min. to transport, 7.5 min. to read, 12.5 min. to print and 25 min. to process. Keeping the process at $\eta = 100\%$ gives a new tape every 25 min., i.e. 12.5 min. average waiting time between batchings. However an overview of the schedule:



reveals that printing and reading would overlap. A corrected schedule:



inserts 12.5 min. transport intervals, so that reading and printing do not collide. This gives $T = 92.5$ min.

Problem 3

If the I/O is controlled by a channel, the reading and printing of a job occurs simultaneous with the processing of another job. Since jobs read and print in less than the 1 minute of CPU required, the job is *CPU bound*. It is possible to set $\eta = 100\%$ and therefore have $T = 60$, as before.

If 1200 cards and 1500 lines are required per job, it now takes 1.2 minutes and 1.5 minutes to read and print a job, respectively. Hence the *bottle-neck* is the printer. The printer sets the lower limit on a time per job at 1.5 minutes. The processor will idle for the .5 minutes the job does not need, and the card reader will idle for .3 minutes. The calculations are now done considering the printer alone:

$$\begin{aligned} T &= 60 \text{ min/hr } \times 1000 \text{ lines/min } \times 1/1500 \text{ jobs/line } = 40 \text{ jobs/hr}, \\ \eta &= 1/1.5 = 66\% \end{aligned}$$

Problem 4

Since the printer cannot keep up with the processor, the buffer will grow without bound if $\eta > 66\%$. If we set $\eta = 66\%$, then we will maintain two print files and two card files all the time. The processor will reading from a card file and writing to a print file, while the print spooler is printing from a print file and the read spooler is reading to a read file. Hence:

$$2 \times (80 \times 1.2K + 100 \times 1.5K) = 492K$$

where K is a kilobyte, a thousand bytes.

Additional memory doesn't really seem to help, unless I get more printers.