Computer Science 340

Operating Systems

TEST

1989

A.

Illustrate what is meant by "off-lining" by describing the sequence of operations needed to process a data file F, presented on punched cards, through a programme P run on a large fast computer to produce a result file R, printed on paper. Call the large fast expensive computer LFE and the small slow cheap computer SSC. Also describe briefly how the job would be handled by a spooling system.

(3 marks)

Write down expressions for the cost of each of the steps in the operations described, using these symbols for the quantities involved :

Size of F :	I kB
Size of R :	O kB
Speed of card reader :	c kB s^{-1}
Speed of line printer :	p kB s^{-1}
Speed of tape drive :	t kB s^{-1}
Processing speed of LFE :	L kB s^{-1}
Running cost of SSC :	S \$ s^{-1}
Running cost of LFE :	R \$ s^{-1}

Assume that :

The "kB" used to calculate the LFE processing time is the sum of the input and output file sizes (I + O).
No time other than input and output time is spent on SSC.
The programme in LFE reads its data directly from the input device and writes its results directly to the output device. (Yes, it's unrealistic, but I don't suppose you want additional complications from disc speeds on LFE.)
There is no overlapping of input, output, and processing.
The speed of the devices is independent of the computer to which they are attached.

(4 marks)

If the device speeds are given by c = p = 0.5 and t = 50, find the condition for off-lining to be cheaper than spooling.

(3 marks)

B.

Explain the meaning of this segment of occam code :

(Recall that a *PAR* construct completes only when *all* its components have completed. The *VAR* line declares two local variables.)

(5 marks)

Sketch an occam programme to implement a circular buffer of 10 characters. The programme should read from channel *in* or write to channel *out* whenever prompted by a message "get" or "put", as appropriate, received on channel *control*. Assume that the buffer *buff* has been declared, and that its elements are named { *buff*[*i*], *i* = 0, 9 }, and that variables *head* and *tail* index the first element in the queue, and the position after the queue, respectively. Ignore the possibilities of buffer overflow or underflow. Explain the function of each statement in your programme.

(5 marks)