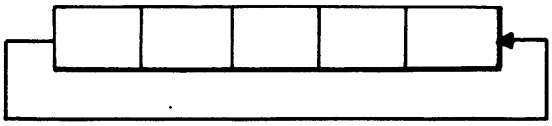

FIGURE 2. Round-robin queue.

(e.g., I-O operation), the entire 4K of user core memory is written out on the swapping drum and some other user program is read into core. This operation is known as "swapping" and it is controlled by the Executive routine called the Swapper. The Swapper administrates the queueing algorithm, making all decisions about when to perform a swap and which user program should be permitted to run next.

One simple queueing algorithm is the "round-robin" method. This algorithm uses a circular queue of users: the user at the top of the queue is given one time quantum and is then placed at the bottom of the queue (see Fig. 2).

The round-robin algorithm, however, fails to account for several problems which occur in actual practice. It does not allow for the fact that before a quantum of time has passed the running user may be "hung" waiting for something to happen. It also ignores the problem of a user losing incoming data because the interval between his time quanta is too long (i.e., there are too many users).

Another disadvantage of the round-robin method is that it makes no attempt to match the amount of time allocated to a user to the past history of his requirements. This is an important consideration, both from the point of view of satisfying the greatest number of users and from the point of view of minimizing the number of swaps performed. A reasonable assumption is that an actual