Consider a simple computer with 64K of physical memory. The OS for this computer uses simple swapping with variable-size partitions. Suppose that at a given point in time, the physical memory of the computer is allocated as shown at right. (Note that the figure is not drawn exactly to scale.)

Assume that during the course of normal operation, the following sequence of requests for new partitions will occur:

- Job 4: 6K
- Job 5: 14K
- Job 6: 8K
- Job 7: 10K
- Job 8: 12K
- Hole: 8K
- Job 1: 8K
- Job 3: 8K
- Hole: 12K
- Job 2: 16K
- Job 0: 12K

For each of the policies below, you are to draw a diagram of the state of memory allocation after each new process has been allocated a partition. If there is not enough space to allocate a partition, you are to remove partitions one at a time in order of the job number (job 0 should be removed first, then job 1, etc.), trying again to allocate the new partition after each removal. Partitions allocated to a hole should fill from the bottom first – any leftover space should appear at the top.

When you are done, you should have 20 diagrams (5 jobs x 4 parts to the problem). Try to draw your diagrams to scale as best you can. (It may help to use graph paper.) Calculate the external fragmentation (as a fraction of total memory) at the end of each simulation (i.e., after job 7 has been allocated its partition).

- a. Best fit
- b. Worst fit
- c. First fit (scan from bottom to top)
- d. Next fit (again, scan upwards, wrapping around as necessary)