

OSLab3 Questions

Will Wainscott

October 2019

1 Explain the difference between internal and external fragmentation.

Internal Fragmentation occurs when a process takes up more space than it needs in memory. This means that parts of the memory will be allocated to a process, but that process does not actually need that extra space. This is common in storage systems that put processes into predefined partitions in the memory. Internal Fragmentation is something that we are going to have happen a lot in our iProject. External Fragmentation occurs when there is parts of the memory that are in between sections of stored processes. These "holes" are not big enough for other processes, but take up more space than the size of potential processes. While there s no process allocated to these sections, they are basically wasted space until a section next to it is cleared up and the hole becomes big enough for a process. This commonly occurs in dynamically stored memory, and is something that we will no encounter in our iProject.

2 Given five (5) memory partitions of 100KB, 500KB, 200KB, 300KB, and 600KB (in that order), how would optimal, first-fit, best-fit, and worst-fit algorithms place processes of 212KB, 417KB, 112KB, and 426KB (in that order)?

First fit:

1. Put the 212KB process into the 500KB partition. Creates another 282KB sized partition.
2. Put the 417KB process into the 600KB partition. Creates another 183KB sized partition.
3. Put the 112KB process into the 282KB sized partition(the one created by putting 212KB into the 500KB partition). Creates another 70KB sized partition.

4. The 426KB process has to wait for either the 417KB process to finish, or both the 212KB and the 112KB process to finish.

Result: This was not very optimal as the 4th process had to wait for an open spot due to external fragmentation.

Best Fit:

1. Put the 212KB process into the 300KB partition. Creates another 88KB sized partition.
2. Put the 417KB process into the 500KB partition. Creates another 83KB sized partition.
3. Put the 112KB process into the 200KB partition. Creates another 88KB sized partition.
4. Put the 426KB process into the 600KB partition. Creates another 184KB sized partition.

Result: This was rather optimal, as every process had a place to go, but we did have to scan the size of the partitions each time before putting a process into one.

Worst Fit:

1. Put the 212KB process into the 600KB sized partition. Creates another 388KB sized partition.
2. Put the 417KB process into the 500KB sized partition. Creates another 83KB sized partition.
3. Put the 112KB process into the 388KB partition (formed by the first process). Creates another 276KB sized partition.
4. The 426KB Process must wait for either the 417KB process to finish or both the 212KB and the 112KB process to finish

Result: This was less than optimal as the 4th process once again had to wait for processes to finish due to external fragmentation.