Name: Nimra Nasir
Reg No: 19-CP-35

Operating System

Lab: 9 (Examples)

Examples:

Example : 1

In the given example, let us assume the jobs and the memory requirements as the following.

Job1 : 90K
Job2 : 20K
Job3 : 50K
Job4 : 200K

free space memory allocation blocks be:

Block1 : 50K
Block2 : 100K
Block 3 : 90k
Block 4 : 200k
Block 5 : 50k

Job 1 (size = 90k)

The first block is not assigned as it is smaller than the demanded memory. 50k (block size) < 90k (job size).

The second block is also not allowed as it will cause memory wastage.

Control moves to the third block which is then allocated as the process size and the memory sizes are equal.

Job1 is assigned to block 3 with no memory wastage.

Job 2 (size = 20K):

The CPU searches the memory for an empty memory slot which is either equal to or more than 100k.

The first block is selected as it is slightly greater than the process size

Block 1 is allocated to the second process.

waste in size is of 30k

Job 3 (size = 50K):

The CPU searches the entire memory block to allocate free memory. The first block is already allocated to job 2. Hence it is skipped. The CPU moves to the second block but is greater in size than the demanded memory (50K < 100k).

The third block is already occupied, and hence skipped

The CPU now moves to the fourth block which is also greater than the demanded size. Hence skipped

Now the fifth block is accessed. In this case, the demanded memory and the available memory sizes are equal (50k = 50k). Hence the block five are allocated to job 3.

Job 4 (size = 200K):

The CPU searches the memory blocks for allocation.

It skips the first blocks as it is already occupied

The second blocks in the memory is skipped as it is smaller than the demanded memory.

The third memory block is also skipped as it is already occupied.

CPU then moves to the fourth memory block which is finally allocated as its size exactly meets the demanded memory size of the process (200k = 200k)

Block four is allocated to the memory which no memory wastage

Example: 2

five memory allocation of 100Kb, 500Kb, 200Kb, 300Kb, 600Kb (in order), how first-fit, best fit would, and worst-fit algorithm place processes of 212Kb, 417Kb, 112Kb and 426Kb (in order). which algorithm makes the most efficient use of memory?

First fit Algorithm:

Memory Partition: 100, 500, 200, 300, 600

Processes: 212, 417, 112, 426

100	500	200	300	600	Memory partition
	212			417	Process

112K is put in 288K (new partition 288K = 500K - 212K)

426K must wait.

Best fit Algorithm:

100	500	200	300	600	Memory partition
	417	112	212	426	Process

Worst fit Algorithm:

100	500	200	300	600	Memory partition
	417			212	Process

112K is put in 388K (new partition 388 = 600 - 212)

426 must wait

Example: 3

These partitions may be allowed in 4 ways.

- First-fit Memory Allocation
- Best-fit Memory Allocation
- worst-fit Memory Allocation
- Next-fit Memory Allocation.

First-fit Memory Allocation:

This method keeps the free / busy list of jobs organized by memory location, low-ordered to high-ordered memory. In this method, first Job claims the first available memory with space more than an or equal to it's size. The operating system doesn't search for appropriate partition but just allocate the job to the nearest memory partition available with sufficient size.

Advantages of first-fit Memory allocation:

It is fast in processing, As the processor allocates the nearest available memory partition to the job, it is very fast in execution.

Disadvantages of first-fit Memory allocation:

It wastes a lot of memory. The processor ignores if the size of partition allocated to the job is very large as compared to the size of job or not. It just allocates the memory. As a result, a lot of memory is wasted and many jobs may not get space in the memory and would have to wait for another job to complete.

Best-fit Memory Allocation:

This method keeps the free / busy list in order by size - smallest to largest. In this method, the operating system first searches the whole of the memory according to the size of the given job and allocates it to the closest-fitting free partition in the memory, making it able to use memory efficiency. Here the jobs are in the order from smallest job to largest job.

Advantages of Best-fit Memory allocation:

Memory Efficient. The operating system allocates the job minimum possible space in the memory, making memory management very efficient. To save memory from getting wasted, it is the best method.

Disadvantages of Best-fit allocation:

It is a slow process. checking the whole memory for each job makes the working of the operating system very slow. It takes a lot of time to complete the work.

Worst-fit Memory Allocation:

In this allocation technique, the process traverses the whole memory and always search for the largest hole/partition, and then the process is placed in the hole/partition, and then the process is placed in that hole/partition. It is a slow process because it has to traverse the entire memory to search the largest hole.

Advantages of worst-fit Allocation:

Since this process chooses the largest hole/partition, therefore there will be large internal fragmentation. Now, this internal fragmentation will be quite big so that other small processes can also be placed in that leftover partition.

Disadvantages of worst-fit Allocation:

It is a slow process chooses because it traverses all the partition in the memory and then selects the largest partition among all the partitions, which is a time consuming process.

Next-fit Memory Allocation:

Next fit is another version of first fit in which memory is searched for empty spaces similar to the first fit memory allocation scheme. Unlike first-fit memory allocation, the only difference b/w the two, is, in the case of next fit. if the search is interrupted in between. the new search is carried out from the last location.

Advantages of Next-fit Allocation:

Short searches (may be shorter than first fit).
Spreads out fragmentation (Like worst fit)
But more fragmentation than best fit.
Guess pointers are a general technique.
Think of them as a lazy (non-coherant) cache.
If they are right, they save a lot of time.
If they are wrong, the algorithm still works
If they can be used in a wide range of problem.

Disadvantages of Next-fit Memory Allocation:

The Bin packed early on may have had room for small items that come later in the list.

Example: 4

Best fit allocates the process to a program which is the smallest sufficient partition among the free available partitions.

Block size = 100, 500, 200, 300, 600.

Process size = 212, 417, 112, 426.

using Best fit Algorithm:

100	500	200	300	600
	417	112	212	426
