**Chapter 4 - Memory**

1. What is the problem with no memory abstraction?

The simplest memory abstraction is no abstraction at all. Early computers did not have any memory abstraction at all, which means all programs had direct access to physical memory. This means we could not have 2 programs running in main memory at once, as this would cause inconsistency in data.

1. What is swapping - batch system?

With a batch system, organizing memory into fixed partitions is simple and effective. Each job is loaded into a partition when it gets to the head of the queue. It stays in memory until it has finished. As long as enough jobs can be kept in memory to keep the CPU busy all the time, there is no reason to use anything more complicated.

3. What are the two methods of memory management?

Memory management techniques, is the method responsible for managing the primary memory in computer memory management function keeps following of the current status in memory location, in case if it’s free or allocated. It measure how memory is allocated over processes, deciding which gets memory, when they receive it, and how much they are free. Processes Scheduling simply is managing the processes residing in the main memory. To express the purpose of scheduling than we can say scheduling forma layout in which the already prioritize processes are loaded into the ready queue of the system and then send for the execution. The scheduling activity usually broken down into three different levels: short, medium, and long -term scheduling. In the following will discuss process, thread, and real-time Scheduling.

4. What are the advantages of the linked list method (Section 4.2.1 & 4.2.2)?\

In computer science, a linked list is a linear collection of data elements whose order is not given by their physical placement in memory. Instead, each element points to the next. It is a data structure consisting of a collection of nodes which together represent a sequence. In its most basic form, each node contains: data, and a reference (in other words, a link) to the next node in the sequence. This structure allows for efficient insertion or removal of elements from any position in the sequence during iteration. More complex variants add additional links, allowing more efficient insertion or removal of nodes at arbitrary positions. A drawback of linked lists is that access time is linear (and difficult to pipeline). Faster access, such as random access, is not feasible. Arrays have better cache locality compared to linked lists.

Linked lists are among the simplest and most common data structures. They can be used to implement several other common abstract data types, including lists, stacks, queues, associative arrays, and S-expressions, though it is not uncommon to implement those data structures directly without using a linked list as the basis.

The principal benefit of a linked list over a conventional array is that the list elements can be easily inserted or removed without reallocation or reorganization of the entire structure because the data items need not be stored contiguously in memory or on disk, while restructuring an array at run-time is a much more expensive operation. Linked lists allow insertion and removal of nodes at any point in the list, and allow doing so with a constant number of operations by keeping the link previous to the link being added or removed in memory during list traversal.

5. Understand algorithms to allocate memory: first fit, next fit, best fit, worst fit (Sectio 4.2.2).

First Fit

In the first fit approach is to allocate the first free partition or hole large enough which can accommodate the process. It finishes after finding the first suitable free partition.

Advantage

Fastest algorithm because it searches as little as possible.

Disadvantage

The remaining unused memory areas left after allocation become waste if it is too smaller. Thus request for larger memory requirement cannot be accomplished.

AD

Best Fit

The best fit deals with allocating the smallest free partition which meets the requirement of the requesting process. This algorithm first searches the entire list of free partitions and considers the smallest hole that is adequate. It then tries to find a hole which is close to actual process size needed.

Advantage

Memory utilization is much better than first fit as it searches the smallest free partition first available.

Disadvantage

It is slower and may even tend to fill up memory with tiny useless holes.

Worst fit

In worst fit approach is to locate largest available free portion so that the portion left will be big enough to be useful. It is the reverse of best fit.

Advantage

Reduces the rate of production of small gaps.

Disadvantage

If a process requiring larger memory arrives at a later stage then it cannot be accommodated as the largest hole is already split and occupied.

AD

Buddy's System

In buddy system, sizes of free blocks are in form of integral power of 2. E.g. 2, 4, 8, 16 etc. Up to the size of memory. When a free block of size 2k is requested, a free block from the list of free blocks of size 2k is allocated. If no free block of size 2k is available, the block of next larger size, 2k+1 is split in two halves called buddies to satisfy the request.

Example

Let total memory size be 512KB and let a process P1, requires 70KB to be swapped in. As the hole lists are only for powers of 2, 128KB will be big enough. Initially no 128KB is there, nor are blocks 256KB. Thus 512KB block is split into two buddies of 256KB each, one is further split into two 128KB blocks and one of them is allocated to the process. Next P2 requires 35KB. Rounding 35KB up to a power of 2, a 64KB block is required.

So when 128KB block is split into two 64KB buddies. Again a process P3(130KB) will be adjusted in the whole 256KB. After satisfying the request in this way when such block is free, the two blocks/buddies can be recombined to form the twice larger original block when it is second half buddy is also free.

Advantage

Buddy system is faster. When a block of size 2k is freed, a hole of 2k memory size is searched to check if a merge is possible, whereas in other algorithms all the hole list must be searched.

Disadvantage

It is often become inefficient in terms of memory utilization. As all requests must be rounded up to a power of 2, a 35KB process is allocated to 64KB, thus wasting extra 29KB causing internal fragmentation. There may be holes between the buddies causing external fragmentation.

Next fit

Next fit is a modified version of first fit. It begins as first fit to find a free partition. When called next time it starts searching from where it left off, not from the beginning.

6. What is the unit of virtual memory, and of physical memory?

7. What is the page table mainly for?

8. What is TLB and what is that for?

9. Differentiate page faults, TLB soft misses and TLB hard misses.

10. What is the essence of PRAs?

**Question for Lab**

1. What is the page table mainly for?
2. What is TLB and what is that for?
3. Differentiate page faults, TLB soft misses and TLB hard misses.
4. A memory free in 4 frames. Which state of the memory after the page 4 is accessed when the requested page as 2 3 2 0 1 5 2 4 5 3 2 5 2 using LRU
5. Assume that the Page Table below is in effect. The number of lines per page is 400. The actual memory location for line 1634 is \_\_34\_\_\_\_ .

|  |  |
| --- | --- |
| Page Number | Page Frame Number |
| 0  1  2  3  4 | 8  10  5  11  0 |