ESPINA, Allain Daniel S. Asst. Prof. Bernard G. Sanidad, DIT, DBA

3ITF September 18, 2023

HOMEWORK 2: MEMORY MANAGEMENT

1. Describe the basic functionality of the tree memory allocation schemes: fixed position partitions, dynamic partitions, and relocatable dynamic partitions.

Fixed (Static) Partitioning, widely recognized as the simplest and oldest memory management schemes, divides the physical memory into fixed portions. Once allocated to a specific process, these partitions will be dedicated to the said process until it is terminated or released. The main concern about using this memory allocation scheme is the high possibility of an internal fragmentation. Internal Fragmentation occurs whenever the requested allocated memory for a process is much larger than the said process, thereby resulting in a waste of memory.

Dynamic (Variable) Partitioning, on the other hand, allocates memory during the run time of a process instead of during the system configuration, with every partition being made just enough for each process. Although this memory allocation scheme solves the problems posed by Fixed Partitioning, it also has its own downside. One particular disadvantage of utilizing this allocation scheme is the occurrence of External Fragmentation, which happens whenever memory is separated by free space, preventing the execution of some processes. Both Fixed and Dynamic Partitioning are types of Contiguous Memory Management Techniques, meaning the memory spaces allocated to a certain process must be adjacent to each other, thus, the occurrence of External Fragmentation in Dynamic Partitioning schemes. In Contiguous Memory Management Techniques, the memory is partitioned into two parts: one storing the operating system itself and the other storing user processes.

Just as the Dynamic Partitioning scheme alleviates some of the shortcomings of Fixed Partitioning, so does the Relocatable Dynamic Partitioning scheme. This memory allocation method utilizes a process called Compaction, wherein the system scans the entire memory for free spaces, all of which will be formed into one memory block that will then be assigned to a specific process. The only downside to this is that the entire operating system must pause all of its tasks to perform compaction, and only then can it resume to its normal flow, which basically means that too much memory compaction may result to excess overhead, and the lack thereof can result into the system being inefficient.

2. Describe best-fit memory allocation and first-fit memory allocation schemes.

Just as its name implies, Best-Fit memory allocation searches through the entire memory for free memory blocks that is closest to the size of a certain process, making it the best fit possible. However, Best-Fit memory allocation runs slower compared to other schemes as it has to traverse the entire memory in search of the smallest free memory block that can hold the process. This scheme is also inefficient to use together with a Fixed Partition scheme as the free memory blocks left by this scheme are so small that it cannot be utilized by any other process, thereby resulting in a waste of space.

Unlike its counterpart, First-Fit memory allocation instead uses the first free block of memory it finds in the memory linked list that can hold the current process, regardless of how much larger it is compared to the process. Once a memory block is selected, two partitions will be made out of it, one of which will be the hole while the other one will be the one to store the process. This scheme is known to be the simplest and easiest to implement among memory allocation schemes, though it also produces the most holes, or the unused memory blocks.

3. Describe how a memory list keeps track of available memory.

Memory monitors and keeps track of used and available memory through the utilization of different memory allocation schemes, like the ones mentioned above. Using these schemes, such as Static, Dynamic, and Relocatable Dynamic addressing, as well as best and first fit practices, enables the system to allocate free memory blocks to the corresponding process which requires it. However, each of these memory allocation schemes have their respective downsides, such as the occurrence of fragmentation, which is why corresponding solutions must be set in place.

4. Discuss the importance of deallocation of memory in a dynamic partition system.

Memory deallocation means that the system pauses or suspends an ongoing process and gives its corresponding memory block to a higher priority process, or simply, the releasing of memory by a finished process. Once the higher priority processes are done, the memory block will be returned to the original process who uses it. This is quite an effective way to free up some memory for higher priority processes to ensure that the entire system can run smoothly with high efficiency.

In the case of Dynamic Partitioning, free memory block sizes are adjusted upon a process' run time, enabling the system to provide just enough free memory to the process. This partitioning scheme ultimately leads to external fragmentation wherein the system will be filled with gaps of free space in between the memory blocks allocated to currently running processes. These "holes" in the memory could possibly be eliminated with the help of memory deallocation which can suspend certain processes to get their memory block and compact it together with other free memory spaces that can be assigned to a process that requires a larger block of memory.

5. Discuss the importance of the bounds register in memory allocation schemes.

The Bounds Register basically sets the higher boundary (Base Register – Lower Boundary) on which memory address is allocated to different processes. This ensures that any process will not try to access any memory addresses outside of the Bounds Register's indicated range. With the help of Bounds Register, these allocation schemes will be aware of which memory spaces are allowed to be allocated to different processes, regardless of if they are free or not, thereby preventing such processes from accessing memory not designated for them.

6. Describe the role of compaction and how it improves memory allocation efficiency.

The Static and Dynamic memory allocation schemes both have their disadvantages being Internal and External Fragmentation, respectively. To provide solution to these concerns, Dynamic Relocatable memory allocation was introduced which utilizes the concept of Compaction to optimize memory usage by collecting all free memory space and combine it into one big block of free memory, which can then be partitioned and assigned to the processes waiting for their corresponding memory block to run.

Although it may seem that Compaction is very useful, it still has its own disadvantages just like other memory allocation schemes. First and foremost, Compaction causes excess overhead which makes the entire system run inefficiently. Apart from that, the system must also pause the operations in order to start the compaction process as the failure to do so may cause even more problems to the system's operations.

References:

BYJU'S. (2022a, October 26). *Dynamic partitioning in operating system: Gate Notes*. BYJUS. https://byjus.com/gate/dynamic-partitioning-in-operating-system-notes/

BYJU'S. (2022b, October 26). *Fixed partitioning in operating system: Gate Notes*. BYJUS. https://byjus.com/gate/fixed-partitioning-in-operating-system-notes/

Dynamic partitioning. Online Courses and eBooks Library. (n.d.-a). https://www.tutorialspoint.com/dynamic-partitioning#:~:text=Operating%20Systems%20%E2%88%92%20Operating%20systems%20use,utilization%20and%20improve%20system%20performance.

GeeksforGeeks. (2021, December 17). *Compaction in operating system*. GeeksforGeeks. https://www.geeksforgeeks.org/compaction-in-operating-system/

GeeksforGeeks. (2023a, February 21). *Difference between internal and external fragmentation*. GeeksforGeeks. https://www.geeksforgeeks.org/difference-between-internal-and-external-fragmentation/

GeeksforGeeks. (2023b, April 24). *Best-fit allocation in operating system*. GeeksforGeeks. https://www.geeksforgeeks.org/best-fit-allocation-in-operating-system/

GeeksforGeeks. (2023c, April 24). *Difference between fragmentation and compaction*. GeeksforGeeks. https://www.geeksforgeeks.org/difference-between-fragmentation-and-compaction/

GeeksforGeeks. (2023d, May 8). *Fixed (or static) partitioning in operating system*. GeeksforGeeks. https://www.geeksforgeeks.org/fixed-or-static-partitioning-in-operating-system/

GeeksforGeeks. (2023e, May 31). *First-fit allocation in operating systems*. GeeksforGeeks. https://www.geeksforgeeks.org/first-fit-allocation-in-operating-systems/

GeeksforGeeks. (2023f, July 28). *Variable (or dynamic) partitioning in operating system*. GeeksforGeeks. https://www.geeksforgeeks.org/variable-or-dynamic-partitioning-in-operating-system/

GeeksforGeeks. (2023g, September 6). *Memory management in operating system*. GeeksforGeeks. https://www.geeksforgeeks.org/memory-management-in-operating-system/

GeeksforGeeks. (2023h, September 10). *Internal fragmentation in os*. GeeksforGeeks. https://www.geeksforgeeks.org/internal-fragmentation-in-os/

GeeksforGeeks. (2023i, September 12). *External fragmentation in os*. GeeksforGeeks. https://www.geeksforgeeks.org/external-fragmentation-in-os/

Goodwin, D. (n.d.). Operating systems - lecture #1: Basic concepts of O/s. https://warwick.ac.uk/fac/sci/physics/research/condensedmatt/imr_cdt/students/david_goodwin/teaching/operating_systems/l1_concepts.pdf

Honarmand, N. (n.d.). Introduction to virtual memory - compas.cs.stonybrook.edu. https://compas.cs.stonybrook.edu/~nhonarmand/courses/fa17/cse306/slides/05-virtual_memory.pdf

Module 29: Operating System Memory Protection - Inflibnet Centre. (n.d.). https://epgp.inflibnet.ac.in/epgpdata/uploads/epgp_content/S000305IT/P001486/M023103/ET/1505719873Module29_operatingsystemmemoryprotection.pdf

Operating system - memory management. Online Courses and eBooks Library. (n.d.-b). https://www.tutorialspoint.com/operating_system/os_memory_management.htm

OS partitioning algorithms - javatpoint. www.javatpoint.com. (n.d.). https://www.javatpoint.com/os-partitioning-algorithms