Operating System

ASSIGNMENT # 02

Submitted By
QASIM ALI (20P-0070)
Submitted to : Sanaa Jeehan
(INSTRUCTOR CS)

DEPARTMENT OF COMPUTER SCIENCE

FAST NATIONAL UNIVERSITY OF COMPUTER AND EMERGING SCIENCES, PESHAWAR

Session 2020-2024

Q1: Why are page sizes always powers of 2?

Ans:
Paging breaks up an address into a page and offset number. It is more efficient to break the address into X page bits and Y offset bits, rather than perform arithmetic on the address to calculate the page number power of 2 in binary, splitting an address between bits results in a page size that is a power of 2. and offset.
You know each bit position represents a Hence
page sizes always powers of 2.

Page sizes are always powers of 2 because it is more efficient to translate a virtual address into a physical address when the page size is a power of 2. This is because the translation can be done with a simple bit mask operation, which is much faster than more complex arithmetic operations.

Q2. Consider a logical address space of 64 pages of 1,024 words each, mapped onto a physical memory of 32 frames.

a. How many bits are there in the logical address?
b. How many bits are there in the physical address?

Ans:

There are 1024 words, 210 = 1024 and 10 bits are needed to represent it.

a.

Page number: Since there are 64 pages, we need enough bits to represent all the page numbers. The number of bits required to represent n pages is log2(n). In this case, log2(64) = 6 bits.

Offset within the page: Since each page contains 1,024 words, we need 10 bits to represent the offset within the page. This is because 2^10 = 1,024.

Result: The logical address space has 64 = 26 pages, hence logical address must be 10+ 6 = 16 bits.

b. To determine the number of bits in the physical address, we need to consider the number of frames in physical memory. Since there are 32 frames, we need enough bits to represent all the frame numbers. The number of bits required to represent n frames is log2(n). In this case, log2(32) = 5 bits.

Result: There are 32 = 25 physical frames, hence physical address are 5 + 10 = 15 bits long.

Q3:

a. What was the maximum size of the swapped-out process?

As allocation happens from left to right, hence while X got its place in memory, there should have been some process to the right of X. In current picture there is an empty fragment of 1MB right of X, hence this should be a process which got swapped out (Max size of swapped out process is 1MB)

b. What was the size of the free block just before it was partitioned by X?

The free block just before it was partitioned by X is the combination of the 2-MB block marked with an X and the 5-MB empty space to its left. Therefore, the size of the free block before partitioning was 2 MB + 5 MB = 7 MB.

C) 3 MB allocation with following strategies?

A new 3-MB allocation request must be satisfied next. Indicate the intervals of memory where a partition will be created for the new process under the following four placement algorithms: best-fit, first-fit, next-fit, worst-fit.

1) Best-fit: The best-fit algorithm allocates the smallest free block that can accommodate the requested size. In this case, the rightmost 3-MB empty space is the best fit for the new 3-MB allocation.

2) First Fit: First block from left to right which satisfy the need. That is right most 4MB vacant slot.
   The first-fit algorithm allocates the first free block encountered that can accommodate the requested size. In this case, the rightmost 4-MB empty space is the first fit for the new 3-MB allocation.

3) Next Fit: Next vacant block after X which have space for 3MB allocation, that is 5MB empty slot

   The next-fit algorithm allocates the next free block encountered after the last allocation, regardless of its size. Since the last allocation was for process X, the next free block is the 5-MB empty space to its right.

4) Worst Fit: The max possible empty slot, that is 8MB
   The worst-fit algorithm allocates the largest free block, leaving smaller blocks for future allocations. In this case, the 8-MB empty space is the worst fit for the new 3-MB allocation.