Implement and simulate fixed partitioning and variable partitioning techniques.

Submitted to : Ms. Pragti Jamwal

AGENDA

- Introduction
- Problem statement
- Theory

INTRODUCTION

Memory partitioning is the system by which the memory of a computer system is divided into sections for use by the resident programs. These memory divisions are known as partitions. There are different ways in which memory can be partitioned: fixed, variable, and dynamic partitioning.

Types of memory partition

Fixed Memory Partition

Variable Memory partition

FIXED MEMORY PARTITION

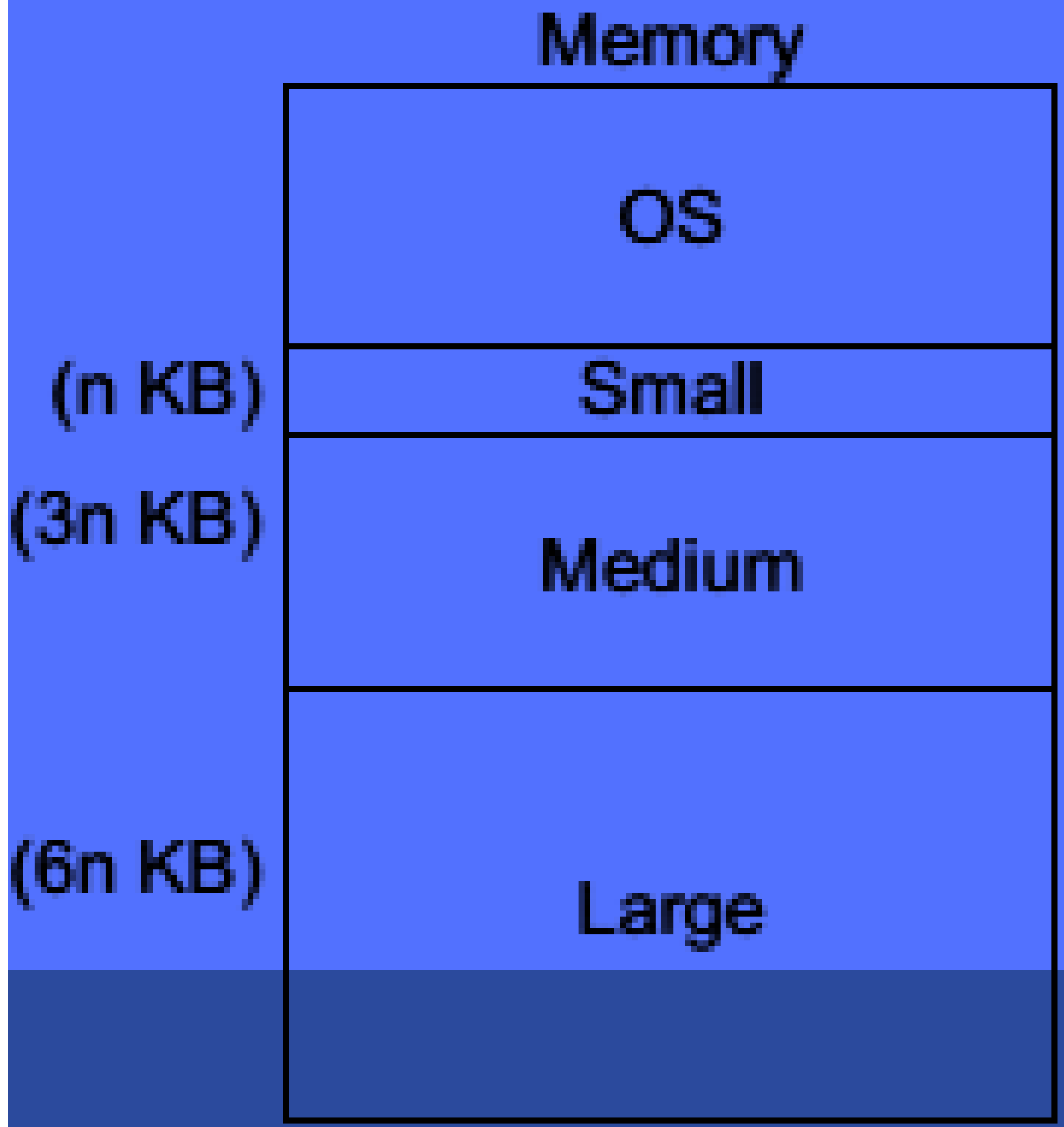
Fixed partitioning is therefore defined as the system of dividing memory into non-overlapping sizes that are fixed, unmoveable, static. A process may be loaded into a partition of equal or greater size and is confined to its allocated partition.

If we have comparatively small processes with respect to the fixed partition sizes, this poses a big problem. This results in occupying all partitions with lots of unoccupied space left. This unoccupied space is known as fragmentation. Within the fixed partition context, this is known as internal fragmentation (IF). This is because of unused space created by a process within its allocated partition (internal).

ALGORITHM

Fixed Partioning Algorithm

- Start the process
- Declare variables
- Enter total memory size
- Allocate Memory for OS
- Allocate total memory to the pages
- Display the wastage of memory
- Stop the process



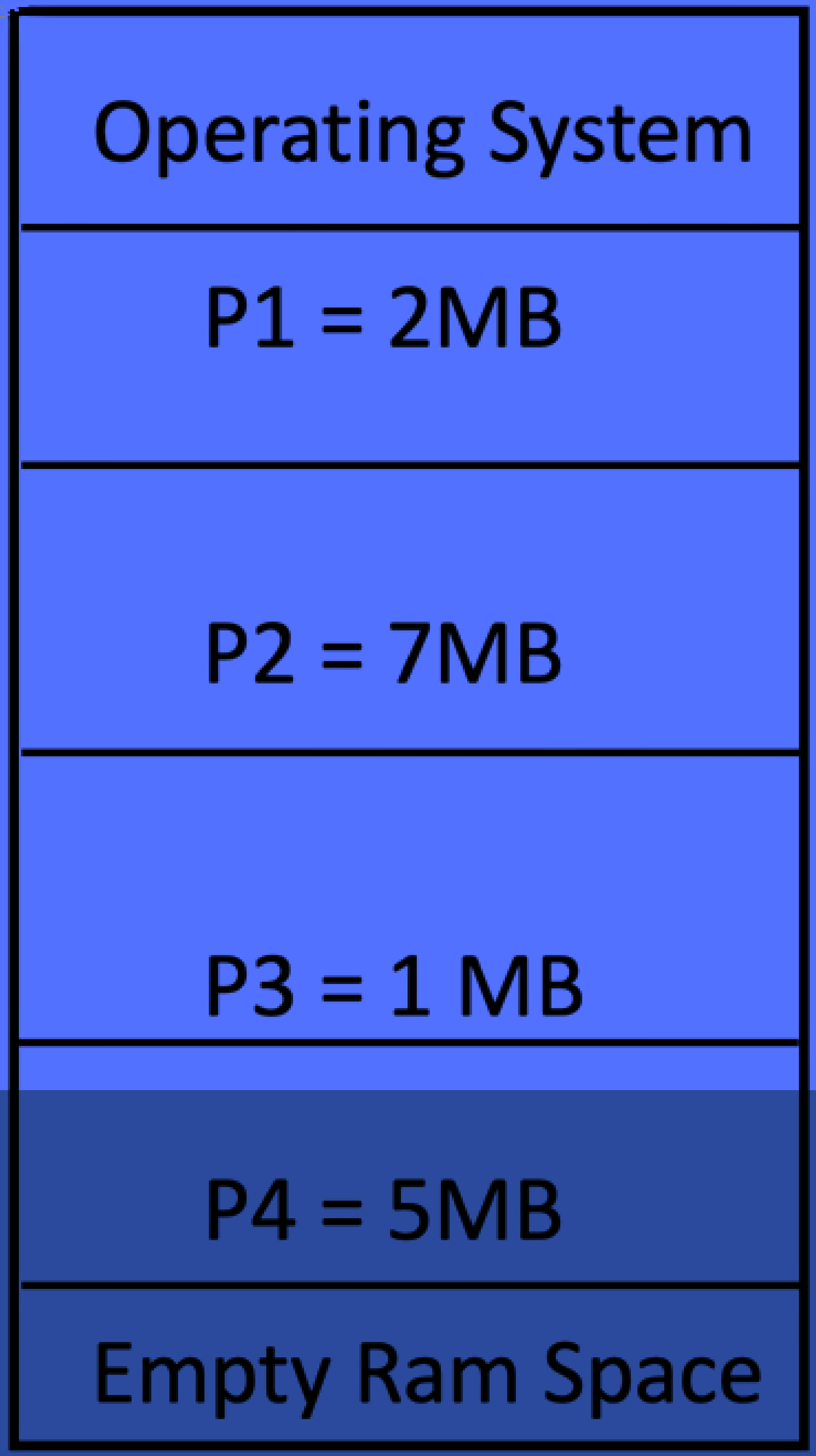
VARIABLE MEMORY PARTITION

Variable partitioning is therefore the system of dividing memory into non-overlapping but variable sizes. This system of partitioning is more flexible than the fixed partitioning configuration, but it's still not the most ideal solution. Small processes are fit into small partitions (item 1) and large processes fit into larger partitions (items 2 and 3). These processes do not necessarily fit exactly, even though there are other unoccupied partitions. Items 3 and 4 are larger processes of the same size, but memory has only one available partition that can fit either of them.
The flexibility offered in variable partitioning still does not completely solve our problems.

ALGORITHM

Variable Partioning Algorithm

- Start the process.
- Declare variables.
- Enter total memory size.
- Allocate memory for os.
- Read the no partition to be divided.
- Read the process no and process size.
- If process size is less than partition size while allocating then, update memory wastage-external fragmentation.
- Print the results



PROBLEM STATEMENT

The fixed-sized partitioning limits the maximum size of processes that are loaded into memory. Even if we have a main memory of 4GB, we cannot load a process of even 10MB if the maximum block size we have is 5MB.

Code for multiprogramming variable task

Code for multiprogramming fixed task

SUMMARY

- Memory partition is of two types - fixed memory and variable memory.
- Fixed memory partition is partition of main memory into a set of non-overlapping memory regions which is fixed.
- Variable memory partition is partition of main memory into a set of non-overlapping memory region which is dynamic.

PRESENTED BY :

1. Ankush Langeh
2. Hitakshi Gupta
3. Ishita Raina
4. Jatanveer Singh
5. Shiven Aggarwal