

Operating system – unit 1

Prepared by Mahesh Kumar

A.P CSE

OPERATING SYSTEM

- Introduction of OS
- Goals and Functionality
- Evaluation of OS
- **Types of OS**
 1. Mainframe Computer
 2. Batch OS Computer
 3. SPOOLING
 4. Multiprogramming
 5. Multitasking

6. Multiprocessing

7. Distributed OS

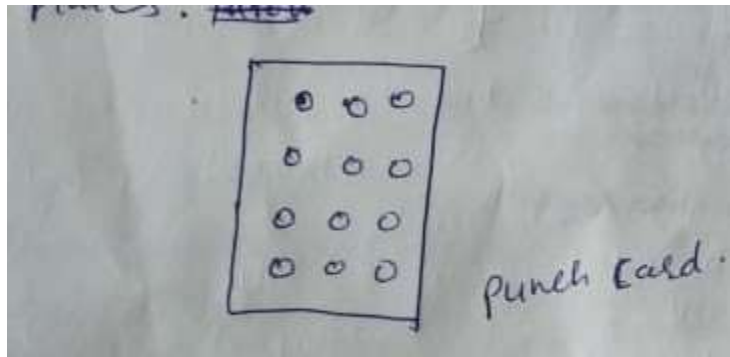
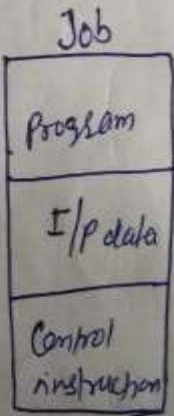
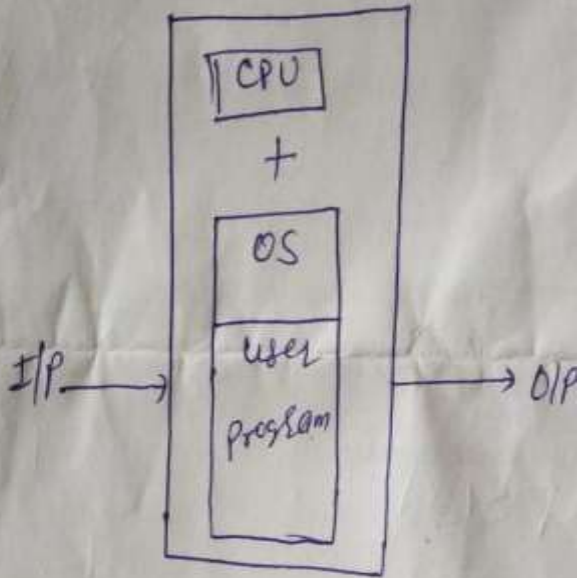
8. Real time OS

9. Network OS

Mainframe Computer

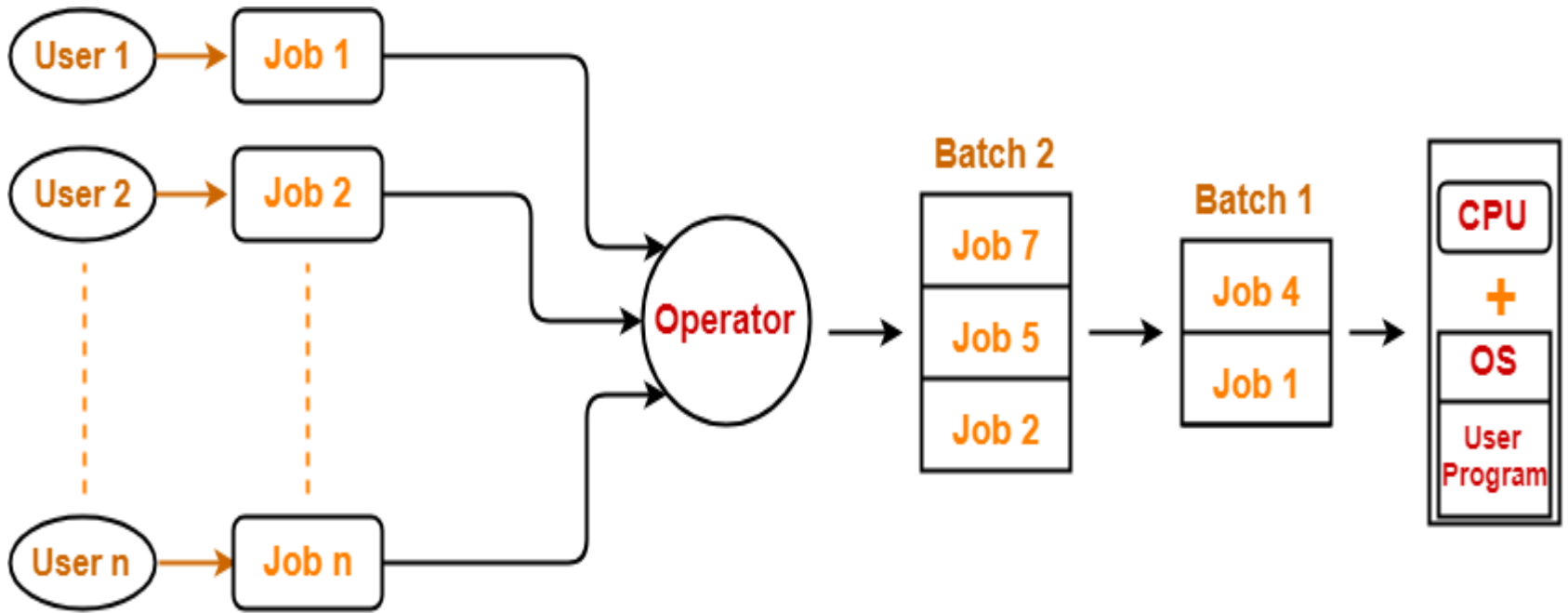


Mainframe Computer



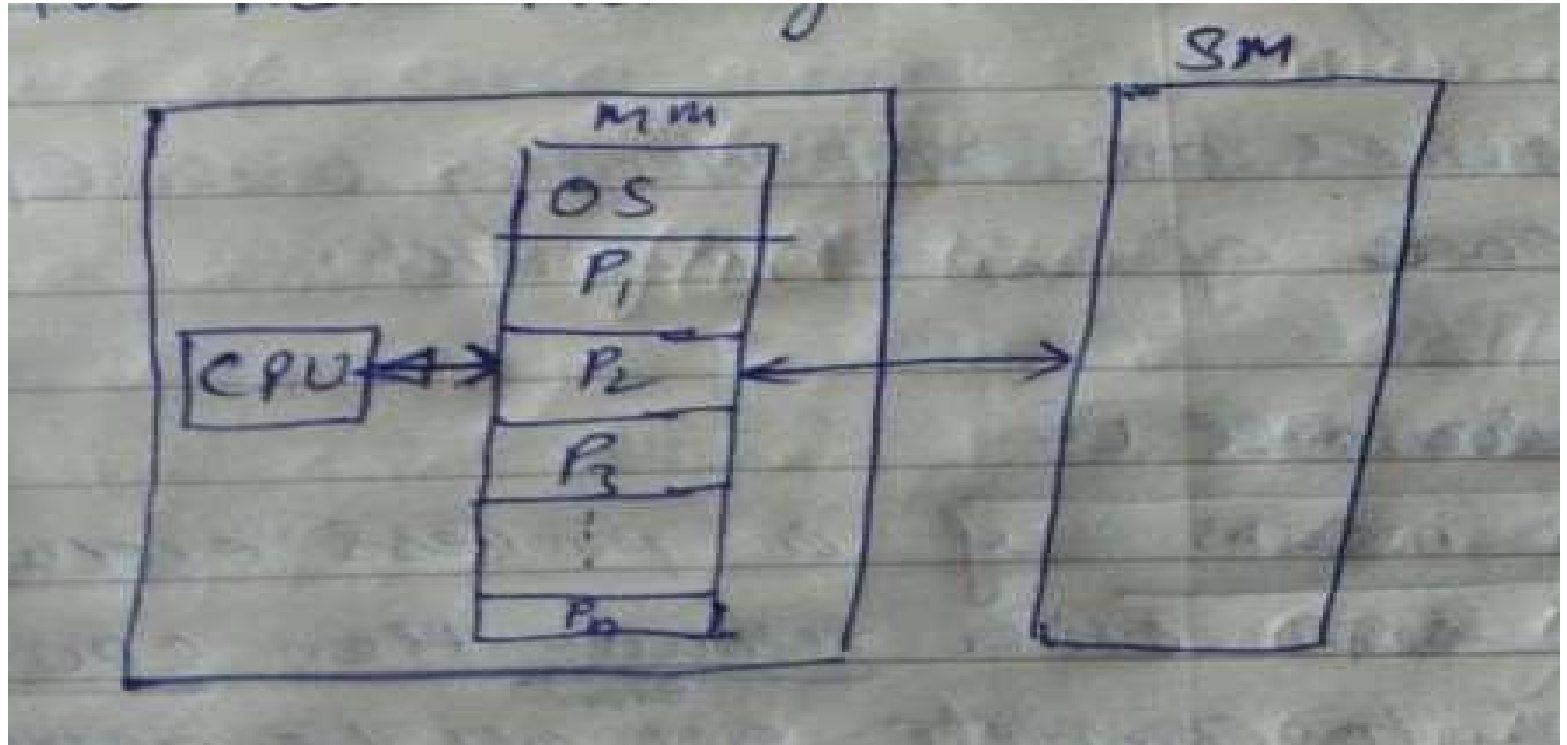
- Computer in broad sense consists of two things
 1. CPU
 2. MEMORY
- They have simple processing unit whose processing is limited not like today.
- Also have limited memory that perform the basic functions i.e. addition, multiplication, division ..etc.
- They don't have memory hierarchy. i.e. cache, main memory,SM.
- They have small cpu with limited memory and a permanent os occupied in main memory.
- Used common input – output device were card reader and a magnetic tape used.
- For input we had punch card with some holes /puncher at the same places as shown in figure . Through these hole light were passed and though others light were not passed.
- Those holes passed the light were considered as logic 1 and from where light were not passed are considered as logic 0.
- Card reader is a device which is used to placed this punch card and plug into the system. similarly for processed i/p were placed at punch card though card reader.
- Some advanced Mainframe computer used magnetics tape for i/p and processed data was also store in magnetic tape.
- User prepared a job which were consists of **Program, input data and control instructions**

Batch operating system



Batch Operating System

Multiprogramming



- To overcome the problem of underutilization of CPU and main memory, the multiprogramming was introduced. The multiprogramming is interleaved execution of multiple jobs by the same [computer](#).
- In multiprogramming system, when one program is waiting for I/O transfer; there is another program ready to utilize the CPU.
- So it is possible for several jobs to share the time of the CPU. But it is important to note that multiprogramming is not defined to be the execution of jobs at the same instance of time.
- Rather it does mean that there are a number of jobs available to the CPU (placed in main memory) and a portion of one is executed then a segment of another and so on.
- A simple process of multiprogramming is shown in figure

Multitasking

- **Multi-tasking** means that the computer can work with more than one program at a time. For instance, you could be working with [information](#) from one [database](#) on the screen analyzing data, while the computer is sorting information from another database, while a spreadsheet is performing calculations on a separate worksheet.
- This is different from "multiple loading" of applications, also known as "context switching" or "task switching," which is what happens when you use *MultiFinder* or *System 7* on the Macintosh or the DOS task switcher.
- Context switching simply allows several applications to be open, but only one is working at a time.

- Actually, even in true multi-tasking, only one application is ever running at anyone instant. But because the computer automatically switches from one to the next so quickly, all the programs *seem* to run simultaneously.
- (With context switching, *you* decide when to shift from one program to another, by hitting a key or clicking the mouse.) Each individual program runs slower, of course, since each gets only a portion of the computer's time, and since some time is lost in the process of going from program to program.
- Multi-tasking capability is built into your computer's operating system. DOS has absolutely no multi-tasking features, but DESQ view and Windows provide it somehow.
- The Mac does allow cooperative multi-tasking in its *background* printing feature, which allows one document to print while you work on another. The word "multi-tasking" is often written without the hyphen; I have kept the hyphen in the word specifically to make it easier to read.

Figure multiprogramming

