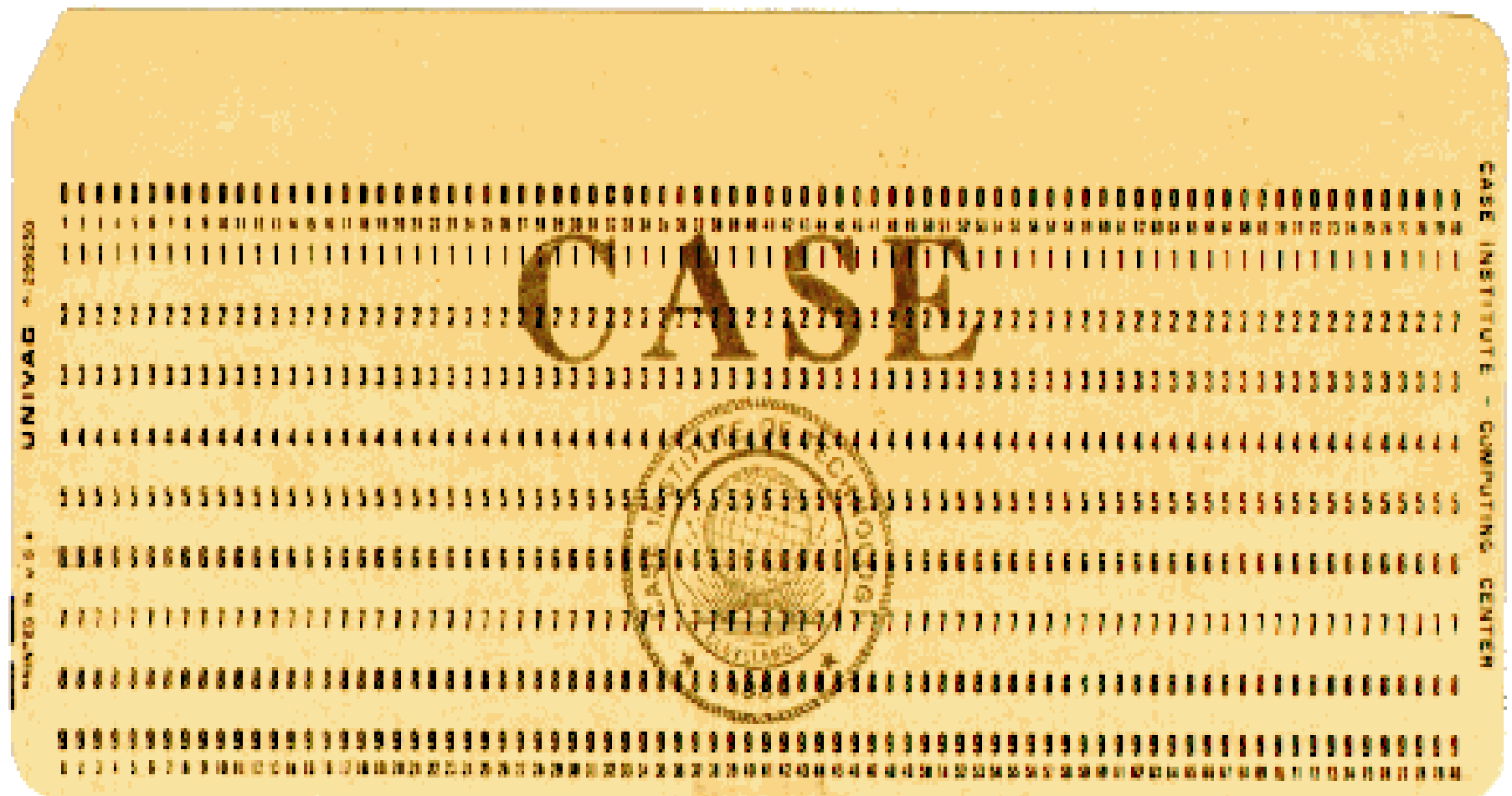


Different Computer Systems

Mainframe Systems

- Are large systems
- Are very expensive
- Early mainframes were **Batch Systems**
- They used punched cards and tape for data and code input
- They used printer for results and error reporting

Punched Card(Hollerith's card)



Mainframe Systems

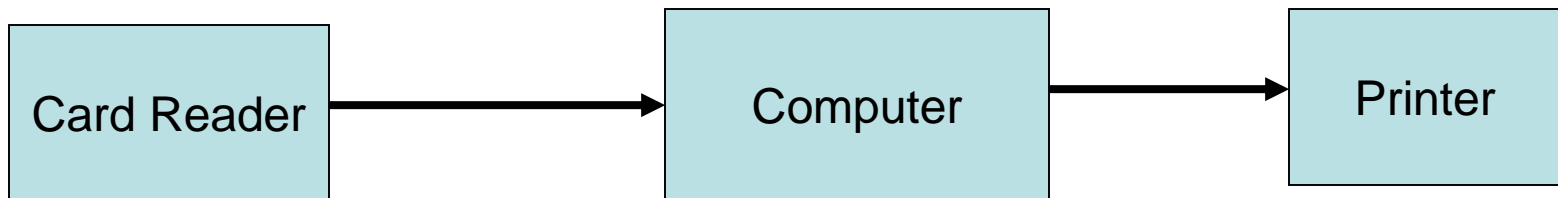
- Reduce setup time by batching similar jobs
- Automatic job sequencing – automatically transfers control from one job to another. **First rudimentary operating system.**
- Resident monitor
 - initial control in monitor
 - control transfers to job
 - when job completes control transfers back to monitor

Batch Systems

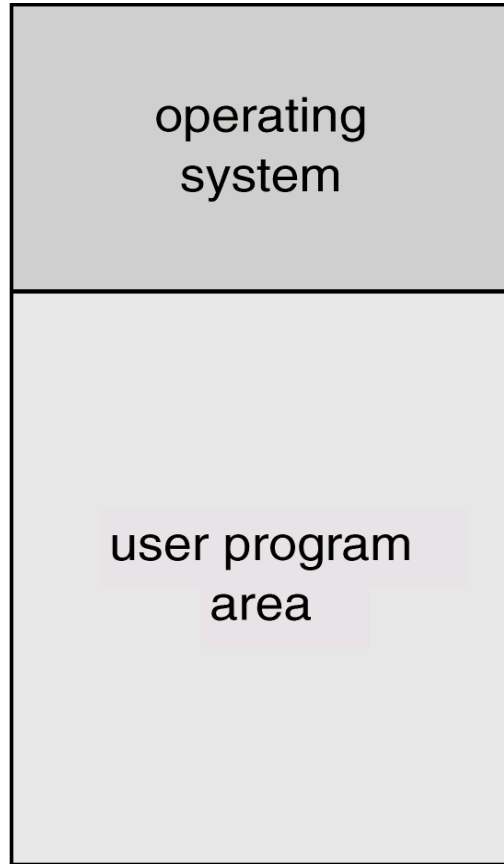
- Users submits job
- Operator forms a batch by bunching user jobs
- Loads job into the card reader
- Computer executes these jobs sequentially and produces results on printer
- Printed output is collected by operator and distributed to users.

Characteristics of Batch System

- Once a job starts executing , until finished no other job can execute
- **Low CPU utilization** because when program is reading data or producing output CPU is idle
- **No user interaction**
- Batch systems are appropriate for executing large jobs that require little interaction

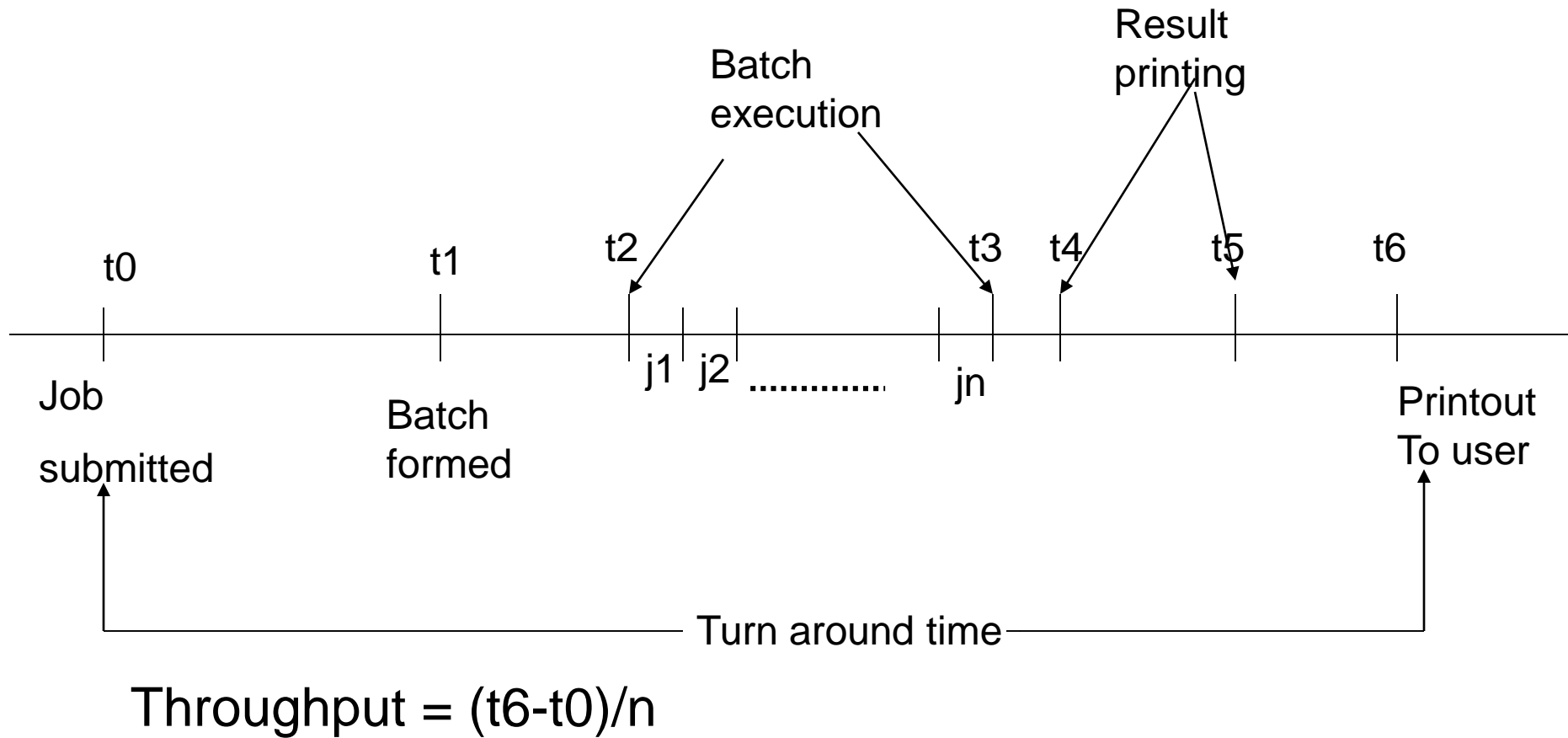


Batch system Memory Layout

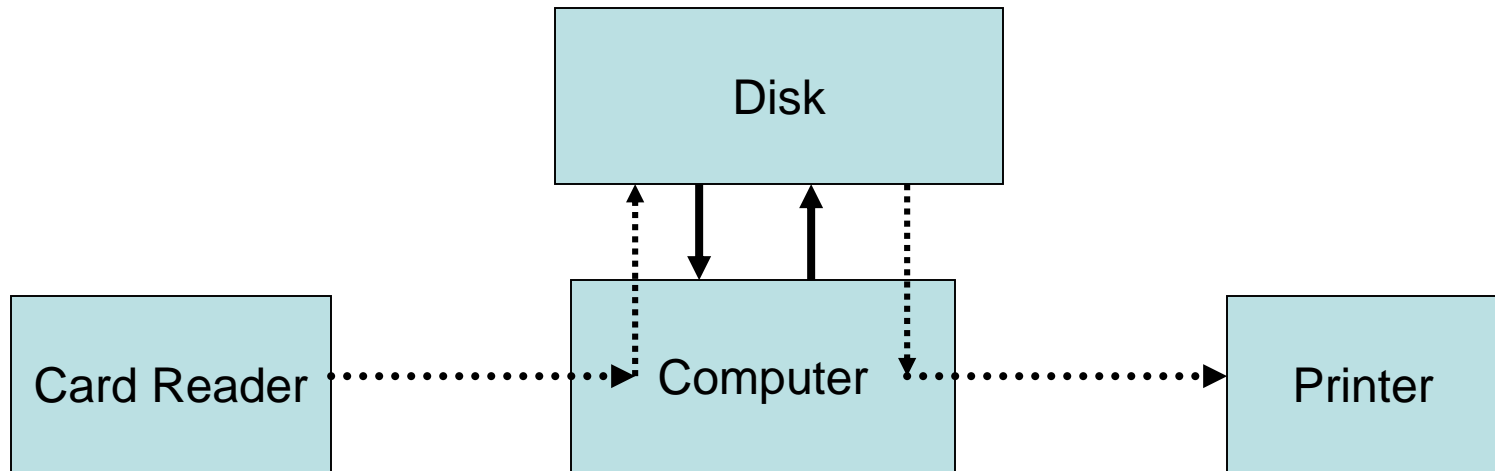


- Main task of the Batch processing OS was to transfer control from one job to another.
 - Problems
 1. How does the monitor know about the nature of the job (e.g., Fortran versus Assembly) or which program to execute?
 2. How does the monitor distinguish
 - (a) job from job?
 - (b) data from program?
 - Solution
 - Job Control Language

Turn around time in batch system



Batch system with spooling



- Every executing program usually will have CPU Burst then I/O followed by CPU Burst and so on.
The last burst is always CPU Burst and then program terminates

Problems of mainframe

- Early days Mainframes were inefficient as execution and I/O was done sequentially
- AS I/O devices are much slower than CPU execution , CPU utilization was poor

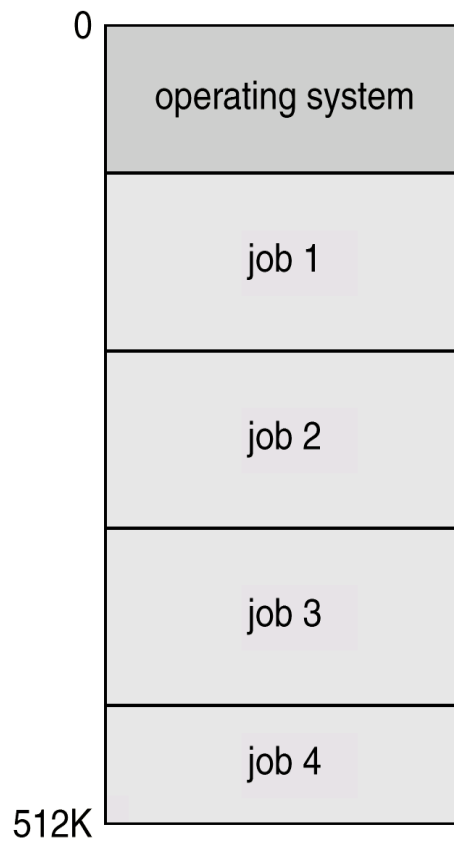
Solution

- Perform execution and I/O in concurrent manner
- Multiprogramming & Time sharing

Multiprogramming

- In this environment, multiple programs are kept in memory and one program is executed at a time. When a program terminates or goes for I/O resources, another program starts executing
- Multiprogramming improves CPU utilization
- Improves throughput

Memory Layout for multiprogramming



OS Features Needed for Multiprogramming

- **Memory management** – the system must allocate the memory to several jobs.
- **CPU scheduling** – the system must choose amongst several jobs ready to run.
- **Allocation of devices.**

Time sharing

- A time unit is divided into small slice and each user can be allocated CPU in round robin manner
- A time shared OS allows many users to share the computer simultaneously
- Time shared OS uses multiprogramming and CPU scheduling to provide each user with a small portion of time shared computer
- Time sharing machine with Interactive I/O devices improves user response time
- Gives illusion that each user has his own machine

Requirements of time shared system

- Needs to keep track of time
- Requires Dynamic Resource allocation
 - To manage memory processes can be swapped out requiring backing store
- Dynamic resource allocation requires elaborate and complicated resource management and access control technique

Present day.....

- PCs
- Parallel and Distributed systems
- Clusters
- Real Time systems