Peoples Empowerment Group

ISB&M SCHOOL OF TECHNOLOGY, NANDE, PUNE

DEPARTMENT OF COMPUTER ENGINEERING

Academic Year 2017-18

Subject: SP&OS **Subject Teacher: Prof. Sheela Bankar**

Class: T.E. computer

1. Explain the main purpose of an operating system?

Operating systems exist for two main purposes. One is that it is designed to make sure a computer system performs well by managing its computational activities. Another is that it provides an environment for the development and execution of programs.

2. What is demand paging?

Demand paging is referred when not all of a process's pages are in the RAM, then the OS brings the missing(and required) pages from the disk into the RAM.

3. What are the advantages of a multiprocessor system?

With an increased number of processors, there is a considerable increase in throughput. It can also save more money because they can share resources. Finally, overall reliability is increased as well.

4. What is a virtual memory?

Virtual memory is a memory management technique for letting processes execute outside of memory. This is very useful especially is an executing program cannot fit in the physical memory.

5.What is a thread? Give some benefits of multithreaded programming.

A thread is a basic unit of CPU utilization. In general, a thread is composed of a thread ID, program counter, register set, and the stack.

benefits of multithreaded programming.

– there is increased responsiveness to the user
– resource sharing within the process
– economy
– utilization of multiprocessing architecture

6. State the main difference between logical from physical address space.

Logical address refers to the address that is generated by the CPU. On the other hand, physical address refers to the address that is seen by the memory unit.

7. Explain the concept of macros.

A macro instruction is a notational convenience for the programmer.

 A macro is an single line abbreviation used for a group of instructions.

 It allows the programmer to write shorthand version of a program (module programming)

 The macro processor replaces each macro invocation (Call) with the corresponding sequence of statements (Definition)

Macro processor

- Recognize macro definitions
- Save the macro definition
- Recognize macro calls
- Expand macro calls

8. What is YACC ?

- **Tool which will produce a parser for a given grammar**.
- YACC (Yet Another Compiler Compiler) is a program designed to compile a LALR(1) grammar and to produce the source code of the syntactic analyzer of the language produced by this grammar
- Input is a grammar (rules) and actions to take upon recognizing a rule
- Output is a C program and optionally a header file of tokens

Input to yacc is divided into three sections.

... definitions ...

%%

... rules ...

%%

... subroutines ...

- **The definitions section consists of:**
  - token declarations .
  - C code bracketed by **“%{“** and **“%}”.**
  - **the rules section consists of**:
    - grammar .
- **the subroutines section** consists of user subroutines

9. What is RR scheduling algorithm?

◦ RR (round-robin) scheduling algorithm is primarily aimed for time-sharing systems. A circular queue is a setup in such a way that the CPU scheduler goes around that queue, allocating CPU to each process for a time interval of up to around 10 to 100 milliseconds.

10. What are necessary conditions which can lead to a deadlock situation in a system?

- Deadlock situations occur when four conditions occur simultaneously in a system:

  Mutual exclusion;

  Hold and Wait

  No preemption

  Circular wait.

11. **Give an example of a Process State.**

New State – means a process is being created

Running – means instructions are being executed

Waiting – means a process is waiting for certain conditions or events to occur

Ready – means a process is waiting for an instruction from the main processor

Terminate – means a process is stopped abruptly

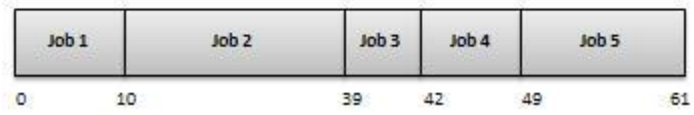
12. **Shown below is the workload for 5 jobs arriving at time zero in the order given below**

Job	**Burst Time**
1	**10**
2	**29**
3	**3**
4	**7**
4	**12**

Now find out which algorithm among FCFS, SJF And Round Robin with quantum 10,

would give the minimum average waiting time.

For FCFS, the jobs will be executed as:



Job	**Waiting Time**
1	0
2	10

3	39
4	42
5	49

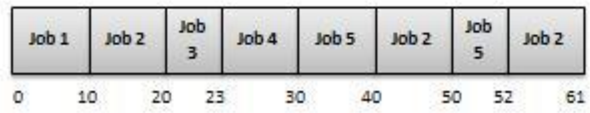
The average waiting time is 140/5=28.

For SJF (non-preemptive), the jobs will be executed as:

Job	**Waiting Time**
1	10
2	32
3	0
4	3
5	20
	65

The average waiting time is 65/5=13.

For Round Robin, the jobs will be executed as:



Job	**Waiting Time**
1	0
2	32
3	20
4	23
5	40
	115

The average waiting time is 115/5=23.

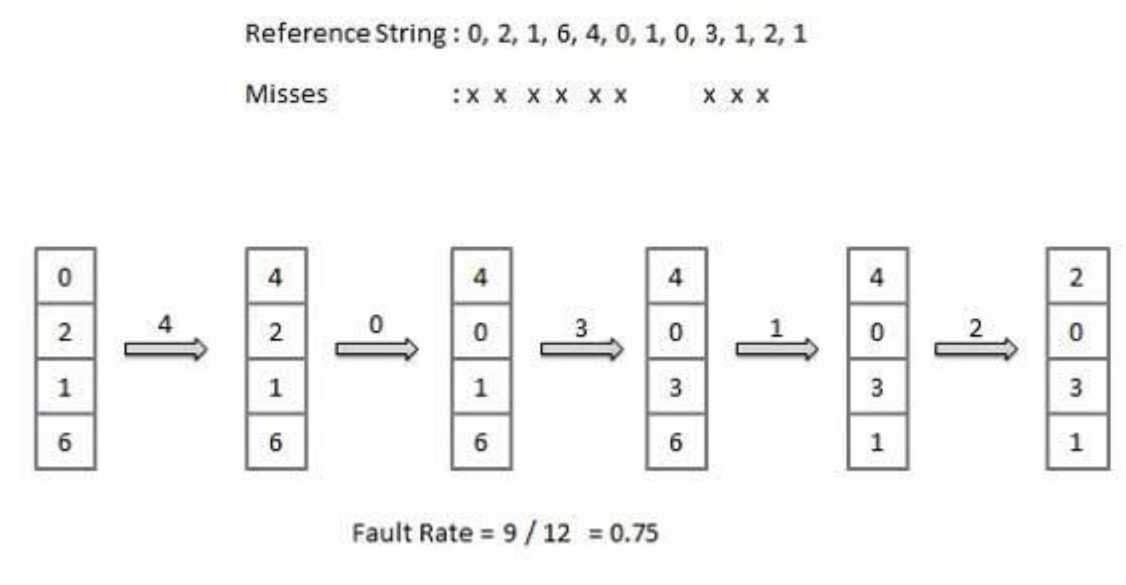
Thus SJF gives the minimum average waiting time.

13. When does a page fault occur?

In demand paging memory management technique, if a page demanded for execution is not present in main memory, then a page fault occurs. To load the page in demand into main memory, a free page frame is searched in main memory and allocated. If no page frame is free, memory manager has to free a frame by swapping its contents to secondary storage and thus make room for the required page. To swap pages, many schemes or strategies are used.

14. Explain FIFO (First in First out) Page Replacement Algorithm.

FIFO (First in First out) Page Replacement Algorithm − It is one of the simplest page replacement algorithm. The oldest page, which has spent the longest time in memory is chosen and replaced. This algorithm is implemented with the help of FIFO queue to hold the pages in memory. A page is inserted at the rear end of the queue and is replaced at the front of the queue.


Reference String : 0, 2, 1, 6, 4, 0, 1, 0, 3, 1, 2, 1
Misses :x x x x x x x x x
0 2 1 6
4
4 2 1 6
0
4 0 1 6
3
4 0 3 6
1
4 0 3 1
2
2 0 3 1
Fault Rate = 9 / 12 = 0.75

In the fig., the reference string is 5, 4, 3, 2, 5, 4, 6, 5, 4, 3, 2, 6 and there are 3 frames empty. The first 3 reference (5, 4, 3) cause page faults and are brought into empty frames. The next reference (2) replaces page 5 because page 5 was loaded first and so on. After 7 page faults, the next reference is 5 and 5 is already in memory so no page fault for this reference. Similarly for next reference 4. The + marks shows incoming of a page while circle shows the page chosen for removal.

Advantages

- FIFO is easy to understand.
- It is very easy to implement.

Disadvantage

- Not always good at performance. It may replace an active page to bring a new one and thus may cause a page fault of that page immediately.

- Another unexpected side effect is the FIFO anomaly or Belady's anomaly. This anomaly says that the page fault rate may increase as the number of allocated page frames increases.

15. Explain NRU(Not Recently Used) Page Replacement Algorithm.

This algorithm requires that each page have two additional status bits 'R' and 'M' called reference bit and change bit respectively. The reference bit(R) is automatically set to 1 whenever the page is referenced. The change bit (M) is set to 1 whenever the page is modified. These bits are stored in the PMT and are updated on every memory reference. When a page fault occurs, the memory manager inspects all the pages and divides them into 4 classes based on R and M bits.

- **Class 1: (0,0)** − neither recently used nor modified - the best page to replace.
- **Class 2: (0,1)** − not recently used but modified - the page will need to be written out before replacement.
- **Class 3: (1,0)** − recently used but clean - probably will be used again soon.
- **Class 4: (1,1)** − recently used and modified - probably will be used again, and write out will be needed before replacing it.

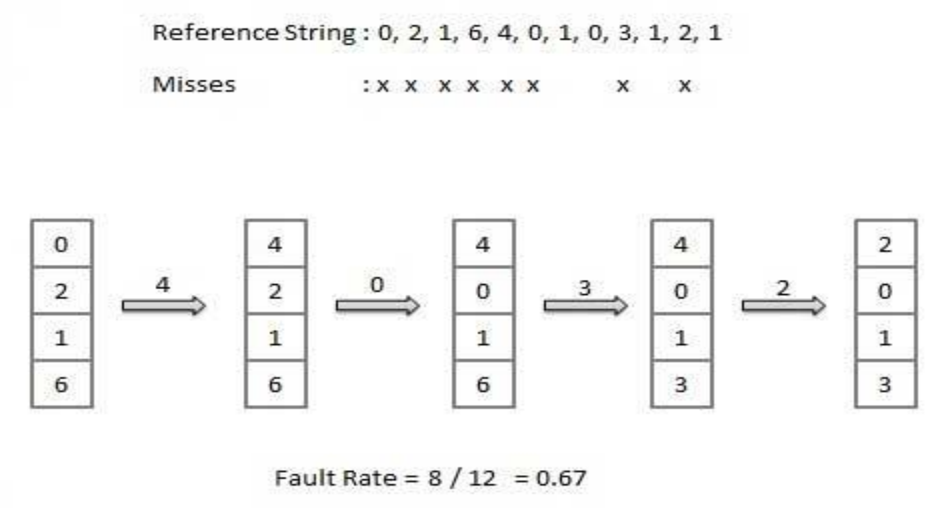
This algorithm removes a page at random from the lowest numbered non-empty class.

Advantages:

- It is easy to understand.
- It is efficient to implement.

16. Explain LRU(Least Recently Used) Algorithm

LRU(Least Recently Used) Algorithm − The Least Recently used (LRU) algorithm replaces the page that has not been used for the longest period of time. It is based on the observation that pages that have not been used for long time will probably remain unused for the longest time and are to be replaced.


Reference String : 0, 2, 1, 6, 4, 0, 1, 0, 3, 1, 2, 1
Misses : x x x x x x x x
0
2
1
6
4
4
2
1
6
0
4
0
1
6
3
4
0
1
3
2
2
0
1
3
Fault Rate = 8 / 12 = 0.67

Initially, 3 page frames are empty. The first 3 references (7, 0, 1) cause page faults and are brought into these empty frames. The next reference (2) replaces page 7. Since next page reference (0) is already in memory, there is no page fault. Now, for the next reference (3), LRU replacement sees that, of the three frames in memory, page 1 was used least recently, and thus is replaced. And thus the process continues.

Advantages

- LRU page replacement algorithm is quiet efficient.
- It does not suffer from Belady's Anomaly.

Disadvantages

- Its implementation is not very easy.
- Its implementation may require substantial hardware assistance.

17. Explain different types of schedulers.

Schedulers are special system software which handle process scheduling in various ways. Their main task is to select the jobs to be submitted into the system and to decide which process to run. Schedulers are of three types −

- Long-Term Scheduler
- Short-Term Scheduler
- Medium-Term Scheduler

Long Term Scheduler

It is also called a **job scheduler**. A long-term scheduler determines which programs are admitted to the system for processing. It selects processes from the queue and loads them into memory for execution. Process loads into the memory for CPU scheduling.

The primary objective of the job scheduler is to provide a balanced mix of jobs, such as I/O bound and processor bound. It also controls the degree of multiprogramming. If the degree of multiprogramming is stable, then the average rate of process creation must be equal to the average departure rate of processes leaving the system.

On some systems, the long-term scheduler may not be available or minimal. Time-sharing operating systems have no long term scheduler. When a process changes the state from new to ready, then there is use of long-term scheduler.

Short Term Scheduler

It is also called as **CPU scheduler**. Its main objective is to increase system performance in accordance with the chosen set of criteria. It is the change of ready state to running state of the process. CPU scheduler selects a process among the processes that are ready to execute and allocates CPU to one of them.

Short-term schedulers, also known as dispatchers, make the decision of which process to execute next. Short-term schedulers are faster than long-term schedulers.

Medium Term Scheduler

Medium-term scheduling is a part of **swapping**. It removes the processes from the memory. It reduces the degree of multiprogramming. The medium-term scheduler is in-charge of handling the swapped out-processes.

A running process may become suspended if it makes an I/O request. A suspended processes cannot make any progress towards completion. In this condition, to remove the process from memory and make space for other processes, the suspended process is moved to the secondary storage. This process is called **swapping**, and the process is said to be swapped out or rolled out. Swapping may be necessary to improve the process mix.

18. Comparison among Scheduler

S.N.	Long-Term Scheduler	Short-Term Scheduler	Medium-Term Scheduler
1	It is a job scheduler	It is a CPU scheduler	It is a process swapping scheduler.
2	Speed is lesser than short term scheduler	Speed is fastest among other two	Speed is in between both short and long term scheduler.
3	It controls the degree of multiprogramming	It provides lesser control over degree of multiprogramming	It reduces the degree of multiprogramming.
4	It is almost absent or minimal in time sharing system	It is also minimal in time sharing system	It is a part of Time sharing systems.
5	It selects processes from pool and loads them into memory for execution	It selects those processes which are ready to execute	It can re-introduce the process into memory and execution can be continued.

19. What are non-preemptive or preemptive algorithms?

Algorithms are either **non-preemptive or preemptive**. Non-preemptive algorithms are designed so that once a process enters the running state; it cannot be preempted until it completes its allotted time, whereas the preemptive scheduling is based on priority where a scheduler may preempt a low priority running process anytime when a high priority process enters into a ready state.

20. Explain Memory Addresses.

Symbolic addresses

The addresses used in a source code. The variable names, constants, and instruction labels are the basic elements of the symbolic address space.

Relative addresses

At the time of compilation, a compiler converts symbolic addresses into relative addresses.

Physical addresses

The loader generates these addresses at the time when a program is loaded into main memory.

Virtual and physical addresses are the same in compile-time and load-time address-binding schemes. Virtual and physical addresses differ in execution-time address-binding scheme.

The set of all logical addresses generated by a program is referred to as a **logical address space**. The set of all physical addresses corresponding to these logical addresses is referred to as a **physical address space.**

The runtime mapping from virtual to physical address is done by the memory management unit (MMU) which is a hardware device. MMU uses following mechanism to convert virtual address to physical address.

- The value in the base register is added to every address generated by a user process, which is treated as offset at the time it is sent to memory. For example, if the base register value is 10000, then an attempt by the user to use address location 100 will be dynamically reallocated to location 10100.
- The user program deals with virtual addresses; it never sees the real physical addresses.