

OPERATING SYSTEM

5CS4 – 03

SOLUTION

## Part - A

### Ans1.

It is a variable or abstract data type used to control access to a common resource by multiple processes in a concurrent system such as a multitasking operating system. A semaphore is simply a variable. This variable is used to solve critical section problems and to achieve process synchronization in the multi processing environment. The two most common kinds of semaphores are counting semaphores and binary semaphores. Counting semaphore can take non-negative integer values and Binary semaphore can take the value 0 & 1 only.

### Ans2.

Inter process communication (IPC) is a mechanism which allows processes to communicate each other and synchronize their actions. The communication between these processes can be seen as a method of co-operation between them. Processes can communicate with each other using these two ways:

1. Shared Memory: A common memory space is shared among all the cooperating process
2. Message passing: The cooperating process communicate via exchanging messages.

### Ans3.

Starvation is a condition where a process does not get the resources it needs for a long time because the resources are being allocated to other processes. It generally occurs in a Priority based scheduling System.

Solution to Starvation: Aging

Aging is a technique of gradually increasing the priority of processes that wait in the system for a long time. For example, if priority range from 127(low) to 0(high), we could increase the priority of a waiting process by 1 Every 15 minutes. Eventually even a process with an initial priority of 127 would take no more than 32 hours for priority 127 process to age to a priority-0 process.

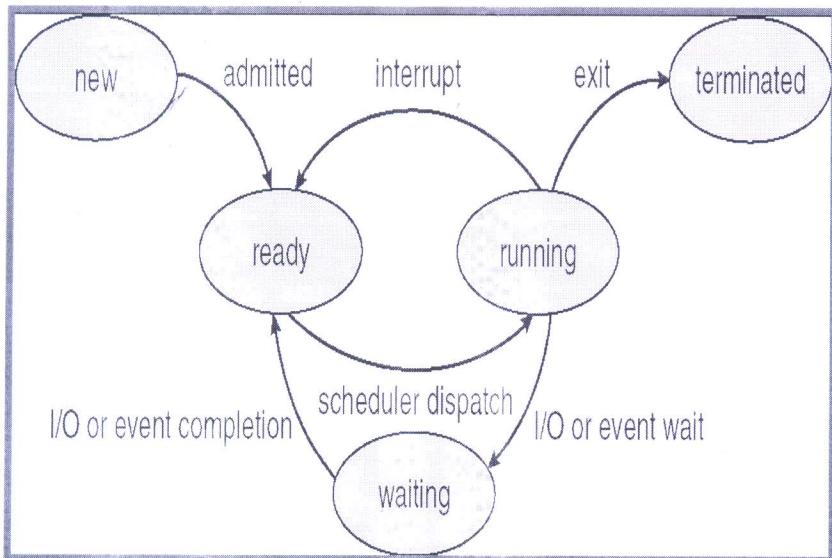
### Ans4.

The main difference between kernel and shell is that the kernel is the core of the operating system that controls all the tasks of the system while the shell is the interface that allows the users to communicate with the kernel.

### Ans5.

A translation lookaside buffer (TLB) is a memory cache that is used to reduce the time taken to access a user memory location. It is a part of the chip's memory-management unit (MMU). The TLB stores the recent translations of virtual memory to physical memory and can be called an address-translation cache.

### Ans6.



### **Ans7.**

The various Disk scheduling methods are:

- i) First Come First Serve
- ii) Shortest Seek Time First
- iii) Scan
- iv) Circular Scan (C – Scan)
- v) Look
- vi) Circular Look (C – Look)

### **Ans8.**

The various attributes of a file are:

- Name . It is the only information which is in human-readable form.
- Identifier. The file is identified by a unique tag(number) within file system.
- Type. It is needed for systems that support different types of files.
- Location. Pointer to file location on device.
- Size. The current size of the file.
- Protection. This controls and assigns the power of reading, writing, executing.
- Time, date, and user identification. This is the data for protection, security, and usage monitoring.

### **Ans9.**

Paging is implemented by breaking up an address into a page and offset number. It is most 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 and offset. Because each bit position represents a power of 2, splitting an address between bits results in a page size that is a power of 2.

### **Ans10.**

There are various operations which can be implemented on a file. We will see all of them in detail.

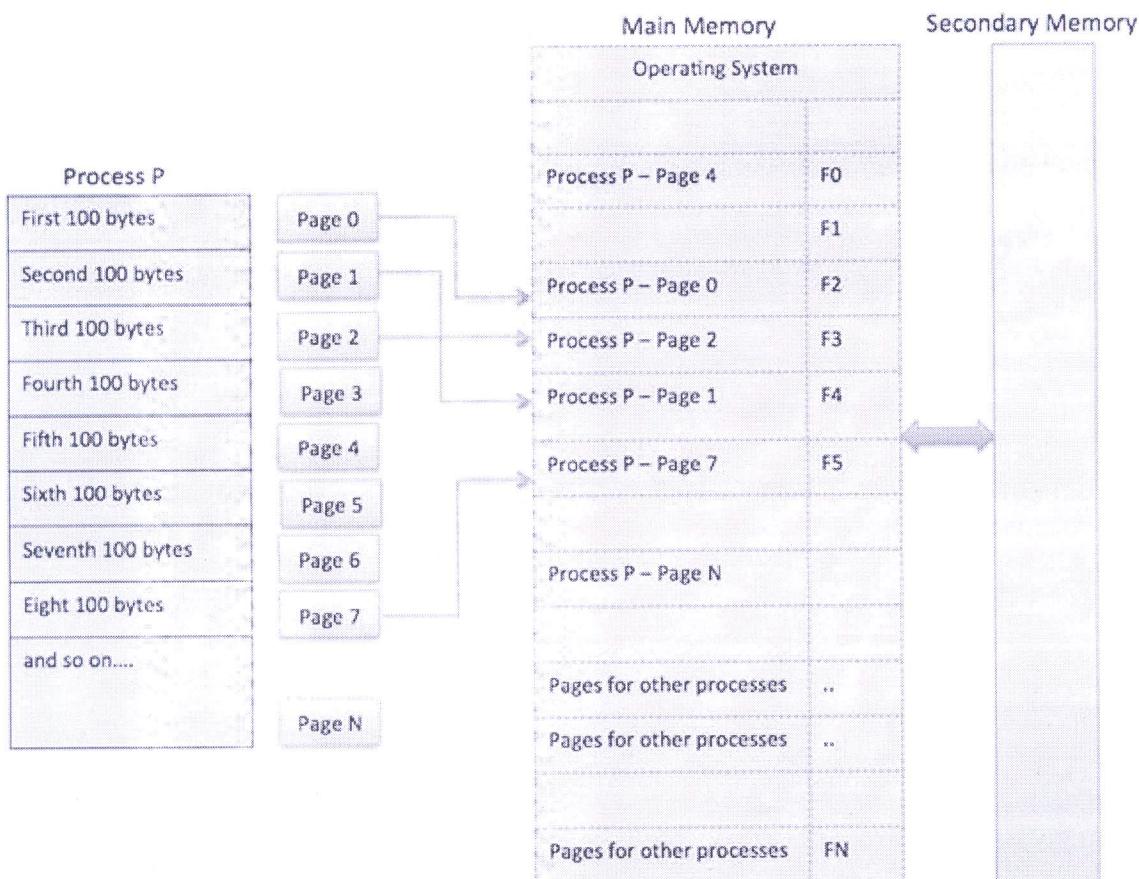
- Create - Creation of the file is the most important operation on the file. Different types of files are created by different methods for example text editors are used to create a text file, word processors are used to create a word file and Image editors are used to create the image files.
- Write - Writing the file is different from creating the file. The OS maintains a write pointer for every file which points to the position in the file from which, the data needs to be written.
- Read - Every file is opened in three different modes : Read, Write and append. A Read pointer is maintained by the OS, pointing to the position up to which, the data has been read.
- Re-position - Re-positioning is simply moving the file pointers forward or backward depending upon the user's requirement. It is also called as seeking.
- Delete - Deleting the file will not only delete all the data stored inside the file, It also deletes all the attributes of the file. The space which is allocated to the file will now become available and can be allocated to the other files.
- Truncate - Truncating is simply deleting the file except deleting attributes. The file is not completely deleted although the information stored inside the file get replaced.

## Part - B

### Ans1.

Paging is a memory management technique in which process address space is broken into blocks of the same size called pages (size is power of 2, between 512 bytes and 8192 bytes). The size of the process is measured in the number of pages.

Similarly, main memory is divided into small fixed-sized blocks of (physical) memory called frames and the size of a frame is kept the same as that of a page to have optimum utilization of the main memory and to avoid external fragmentation.



### Address Translation

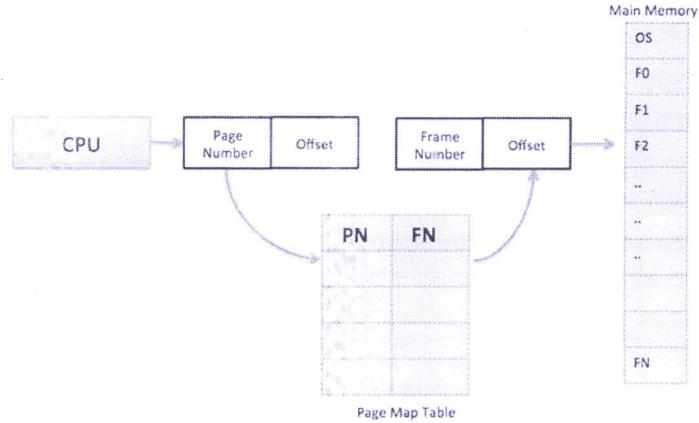
Page address is called logical address and represented by page number and the offset.

Logical Address = Page number + page offset

Frame address is called physical address and represented by a frame number and the offset.

Physical Address = Frame number + page offset

A data structure called page map table is used to keep track of the relation between a page of a process to a frame in physical memory.

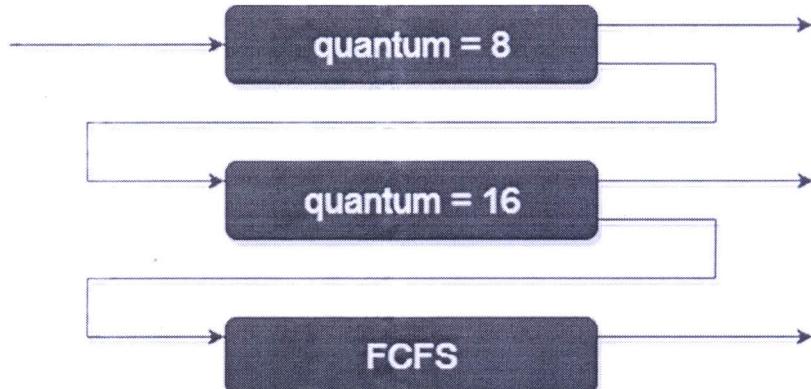


When the system allocates a frame to any page, it translates this logical address into a physical address and create entry into the page table to be used throughout execution of the program.

### Ans2.

In a multilevel queue-scheduling algorithm, processes are permanently assigned to a queue on entry to the system. Processes do not move between queues. This setup has the advantage of low scheduling overhead, but the disadvantage of being inflexible.

Multilevel feedback queue scheduling, however, allows a process to move between queues. The idea is to separate processes with different CPU-burst characteristics. If a process uses too much CPU time, it will be moved to a lower-priority queue. Similarly, a process that waits too long in a lower-priority queue may be moved to a higher-priority queue.



In general, a multilevel feedback queue scheduler is defined by the following parameters:

- The number of queues.
- The scheduling algorithm for each queue.
- The method used to determine when to upgrade a process to a higher-priority queue.
- The method used to determine when to demote a process to a lower-priority queue.
- The method used to determine which queue a process will enter when that process needs service.

The definition of a multilevel feedback queue scheduler makes it the most general CPU-scheduling algorithm. It can be configured to match a specific system under design.

### Ans3.

The various methods to handle deadlocks are:

#### I. Deadlock Detection

Deadlock can be detected by the resource scheduler as it keeps track of all the resources that are allocated to different processes. After a deadlock is detected, it can be handled using the given methods:

- All the processes that are involved in the deadlock are terminated. This approach is not that useful as all the progress made by the processes is destroyed.
- Resources can be preempted from some processes and given to others until the deadlock situation is resolved.

## 2. Deadlock Prevention

It is important to prevent a deadlock before it can occur. So, the system checks each transaction before it is executed to make sure it does not lead to deadlock. If there is even a slight possibility that a transaction may lead to deadlock, it is never allowed to execute.

## 3. Deadlock Avoidance

It is better to avoid a deadlock rather than take measures after the deadlock has occurred. The wait for graph can be used for deadlock avoidance. This is however only useful for smaller databases as it can get quite complex in larger databases.

## 4. Ostrich Algorithm

The ostrich algorithm means that the deadlock is simply ignored and it is assumed that it will never occur. This is done because in some systems the cost of handling the deadlock is much higher than simply ignoring it as it occurs very rarely. So, it is simply assumed that the deadlock will never occur and the system is rebooted if it occurs by any chance.

Deadlock recovery is possible in two ways -

- Process Termination
  - Abort all deadlocked processes:
    - Fast
    - A lot of process work is lost.
  - Abort one deadlocked process at a time and check for deadlocks again:
    - More work to resolve a deadlock.
    - Better in terms of process work.
    - What is a good order to abort processes?
- Resource Preemption
  - what is a good way to select a victim
  - How can we rollback and then recover from preemption?
  - How can we protect from starvation

## Ans4.

The methods by which we can access a file are:

Sequential access

- A sequential access is that in which the records are accessed in some sequence
- The information in the file is processed in order, one record after the other.
- This access method is the most primitive one. Example: Compilers usually access files in this fashion.

Direct/Random access

- Random access file organization provides, accessing the records directly.
- Each record has its own address on the file with the help of which it can be directly accessed for reading or writing.
- The records need not be in any sequence within the file and they need not be in adjacent locations on the storage medium.

## Indexed sequential access

- This mechanism is built up on base of sequential access.
- An index is created for each file which contains pointers to various blocks.
- Index is searched sequentially and its pointer is used to access the file directly.

## Ans5.

The services provided by an Operating System are:

- User Interface

Usually Operating system comes in three forms or types. Depending on the interface their types have been further subdivided. The command line interface (CLI) usually deals with using text commands and a technique for entering those commands. The batch interface (BI): commands and directives are used to manage those commands that are entered into files and those files get executed. Another type is the graphical user interface (GUI): which is a window system with a pointing device to point to the I/O, choose from menus driven interface and to make choices viewing from a number of lists and a keyboard to entry the texts.

- Program Execution in Operating System

The operating system must have the capability to load a program into memory and execute that program. Furthermore, the program must be able to end its execution, either normally or abnormally / forcefully.

- File System Manipulation in Operating System

Programs need has to be read and then write them as files and directories. File handling portion of operating system also allows users to create and delete files by specific name along with extension, search for a given file and / or list file information. Some programs comprise of permissions management for allowing or denying access to files or directories based on file ownership.

- I/O operations in Operating System

A program which is currently executing may require I/O, which may involve file or other I/O device. For efficiency and protection, users cannot directly govern the I/O devices. So, the OS provide a means to do I/O Input / Output operation which means read or write operation with any file.

- Communication System of Operating System

Process needs to swap over information with other process. Processes executing on same computer system or on different computer systems can communicate using operating system support. Communication between two processes can be done using shared memory or via message passing.

- Resource Allocation of Operating System

When multiple jobs running concurrently, resources must need to be allocated to each of them. Resources can be CPU cycles, main memory storage, file storage and I/O devices. CPU scheduling routines are used here to establish how best the CPU can be used.

- Error Detection

Errors may occur within CPU, memory hardware, I/O devices and in the user program. For each type of error, the OS takes adequate action for ensuring correct and consistent computing.

- Accounting

This service of the operating system keeps track of which users are using how much and what kinds of computer resources have been used for accounting or simply to accumulate usage statistics.

- Protection and Security

Protection includes in ensuring all access to system resources in a controlled manner. For making a system secure, the user needs to authenticate him or her to the system before using (usually via login ID and password).

### **Ans6.**

A page fault is a type of exception raised by computer hardware.

A page fault occurs when a program attempts to access a block of memory that is not stored in the physical memory, or RAM. The fault notifies the operating system that it must locate the data in virtual memory, then transfer it from the storage device, such as an HDD or SSD, to the system RAM.

When page fault occurs then following sequence of events happens :

1. The computer hardware traps to the kernel and program counter (PC) is saved on the stack. Current instruction state information is saved in CPU registers.
2. An assembly program is started to save the general registers and other volatile information to keep the OS from destroying it.
3. Operating system finds that a page fault has occurred and tries to find out which virtual page is needed. Some times hardware register contains this required information. If not, the operating system must retrieve PC, fetch instruction and find out what it was doing when the fault occurred.
4. Once virtual address caused page fault is known, system checks to see if address is valid and checks if there is no protection access problem.
5. If the virtual address is valid, the system checks to see if a page frame is free. If no frames are free, the page replacement algorithm is run to remove a page.
6. If frame selected is dirty, page is scheduled for transfer to disk, context switch takes place, fault process is suspended and another process is made to run until disk transfer is completed.
7. As soon as page frame is clean, operating system looks up disk address where needed page is, schedules disk operation to bring it in.
8. When disk interrupt indicates page has arrived, page tables are updated to reflect its position, and frame marked as being in normal state.
9. Faulting instruction is backed up to state it had when it began and PC is reset. Faulting is scheduled, operating system returns to routine that called it.
10. Assembly Routine reloads register and other state information, returns to user space to continue execution.

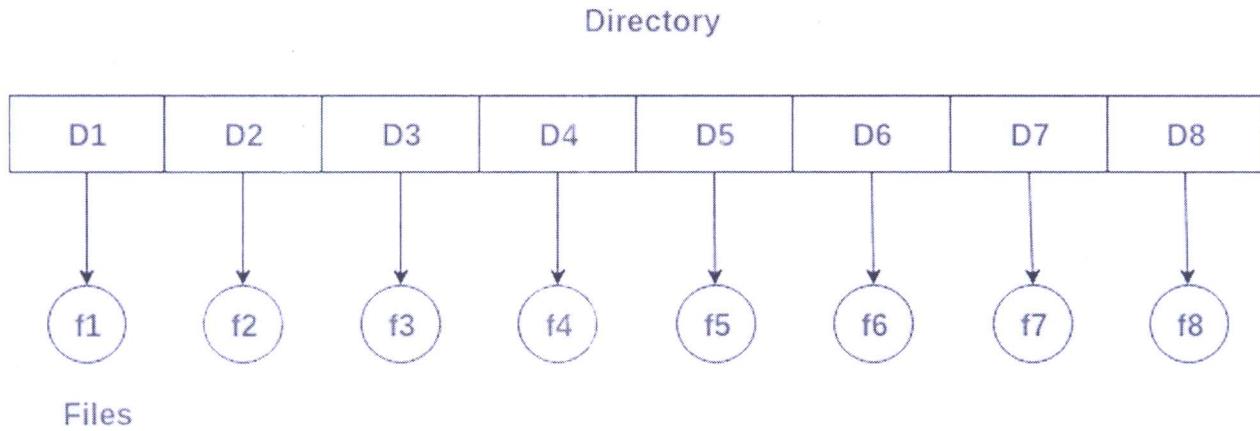
### **Ans7.**

A directory is a container that is used to contain folders and file. It organizes files and folders into a hierarchical manner.

There are several logical structures of a directory, these are given below.

#### **Single-level directory**

Single level directory is simplest directory structure. In it all files are contained in same directory which make it easy to support and understand. A single level directory has a significant limitation, however, when the number of files increases or when the system has more than one user. Since all the files are in the same directory, they must have the unique name . if two users call their dataset test, then the unique name rule violated.



Advantages:

- Since it is a single directory, so its implementation is very easy.
- If files are smaller in size, searching will faster.
- The operations like file creation, searching, deletion, updating are very easy in such a directory structure.

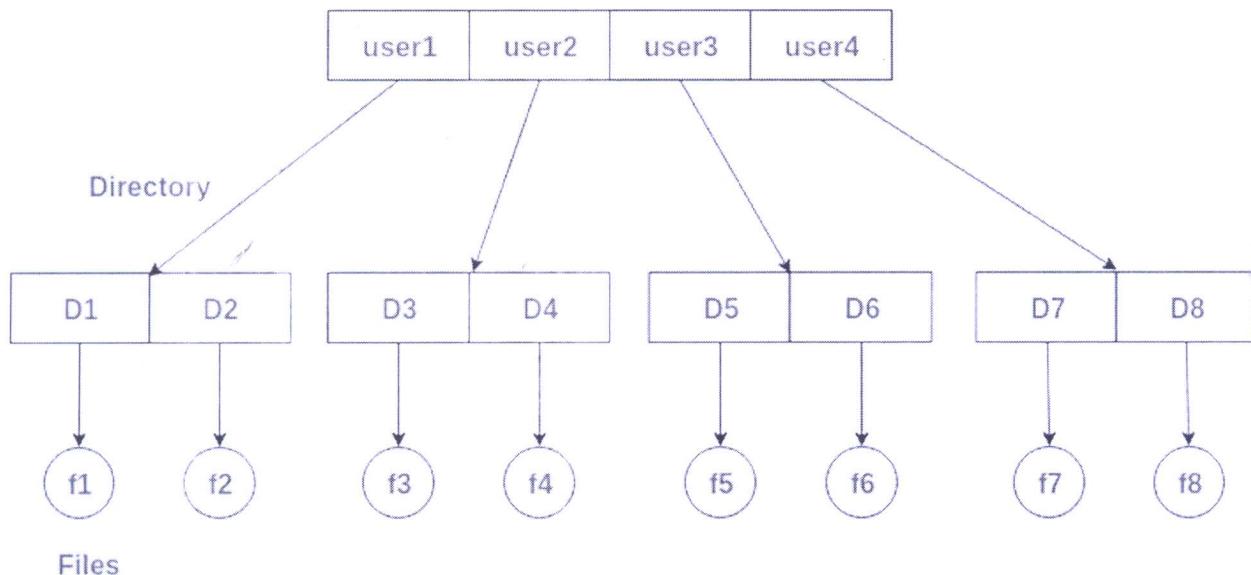
Disadvantages:

- There may chance of name collision because two files can not have the same name.
- Searching will become time taking if directory will large.
- In this can not group the same type of files together.

#### Two-level directory

As we have seen, a single level directory often leads to confusion of files names among different users. the solution to this problem is to create a separate directory for each user.

In the two-level directory structure, each user has their own *user files directory (UFD)*. The UFDs have similar structures, but each lists only the files of a single user. system's *master file directory (MFD)* is searched whenever a new user logs in. The MFD is indexed by username or account number, and each entry points to the UFD for that user.



Advantages:

- We can give full path like /User-name/directory-name/.
- Different users can have same directory as well as file name.
- Searching of files become more easy due to path name and user-grouping.

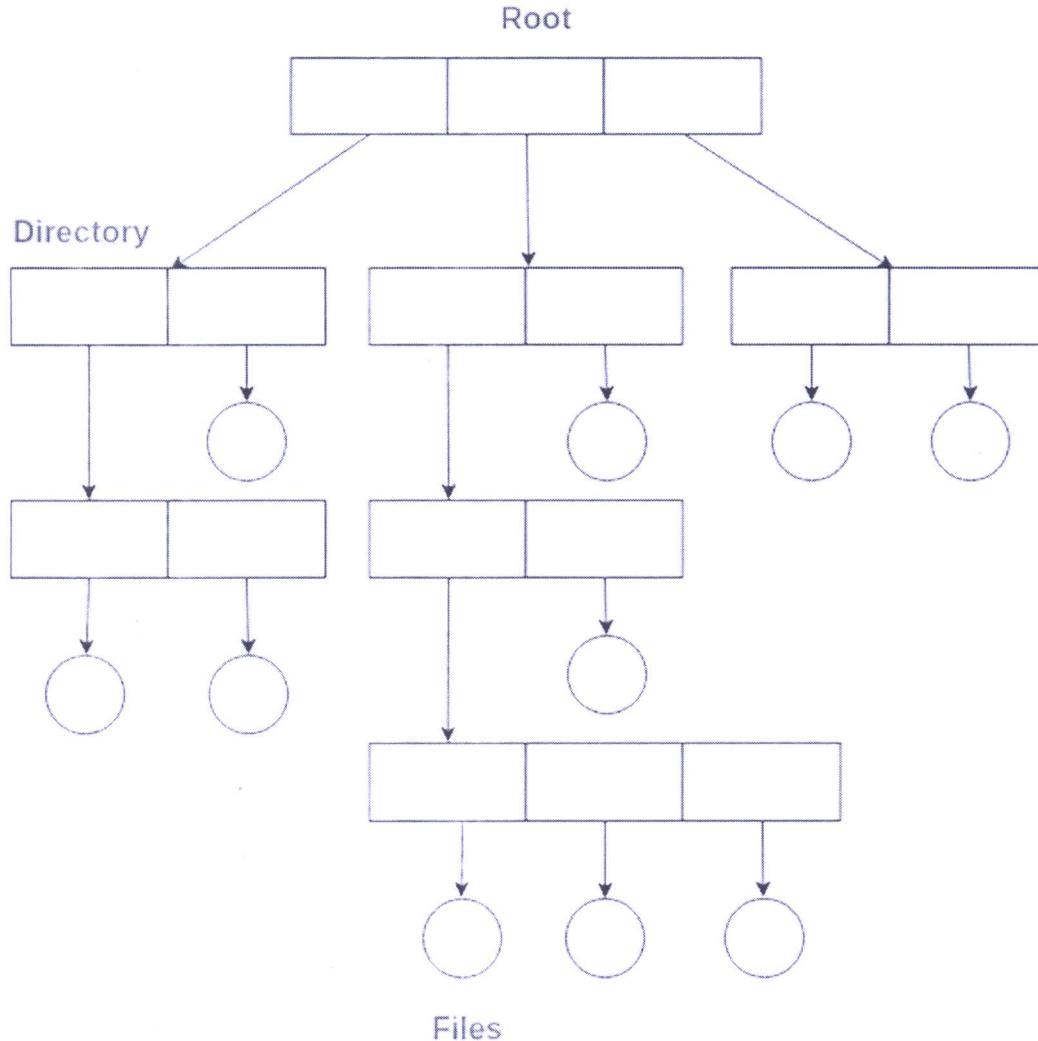
### Disadvantages:

- A user is not allowed to share files with other users.
- Still it is not very scalable, two files of the same type cannot be grouped together in the same user.

### Tree-structured directory

Once we have seen a two-level directory as a tree of height 2, the natural generalization is to extend the directory structure to a tree of arbitrary height.

This generalization allows the user to create their own subdirectories and to organize their files accordingly.



A tree structure is the most common directory structure. The tree has a root directory, and every file in the system has a unique path.

### Advantages:

- Very generalized, since full path name can be given.
- Very scalable, the probability of name collision is less.
- Searching becomes very easy, we can use both absolute path as well as relative.

### Disadvantages:

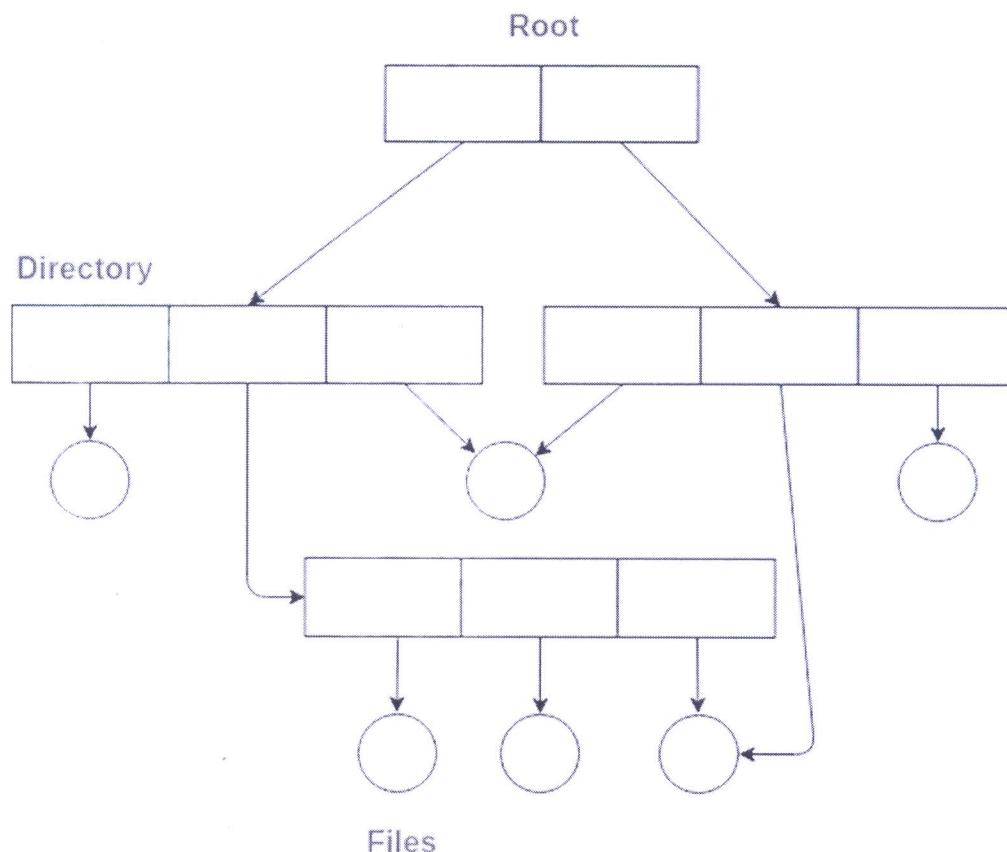
- Every file does not fit into the hierarchical model, files may be saved into multiple directories.
- We can not share files.
- It is inefficient, because accessing a file may go under multiple directories.

## Acyclic-graph-directory

An acyclic graph is a graph with no cycle and allows to share subdirectories and files. The same file or subdirectories may be in two different directories. It is a natural generalization of the tree-structured directory.

It is used in the situation like when two programmers are working on a joint project and they need to access files. The associated files are stored in a subdirectory, separated them from other projects and files of other programmers since they are working on a joint project so they want to the subdirectories into their own directories. The common subdirectories should be shared. So here we use Acyclic directories.

It is the point to note that shared file is not the same as copy file if any programmer makes some changes in the subdirectory it will reflect in both subdirectories.



### Advantages:

- We can share files.
- Searching is easy due to different-different paths.

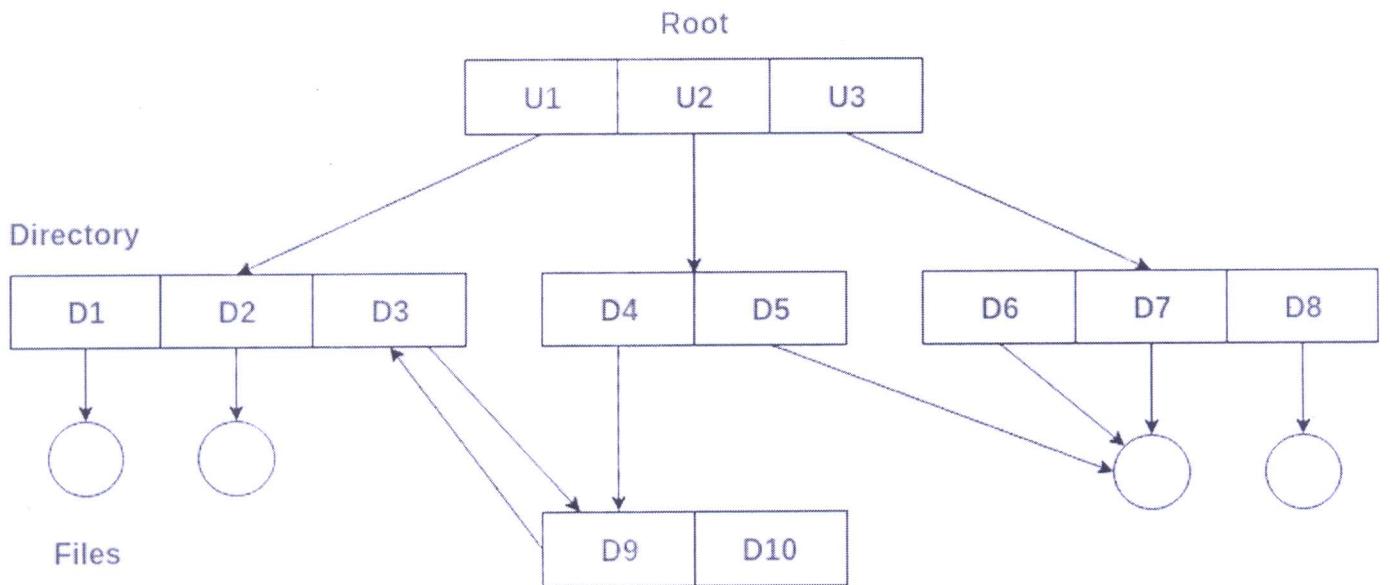
### Disadvantages:

- We share the files via linking, in case of deleting it may create the problem,
- If the link is softlink then after deleting the file we left with a dangling pointer.
- In case of hardlink, to delete a file we have to delete all the reference associated with it.

## General-graph-directory-structure

In general graph directory structure, cycles are allowed within a directory structure where multiple directories can be derived from more than one parent directory.

The main problem with this kind of directory structure is to calculate total size or space that has been taken by the files and directories.



**Advantages:**

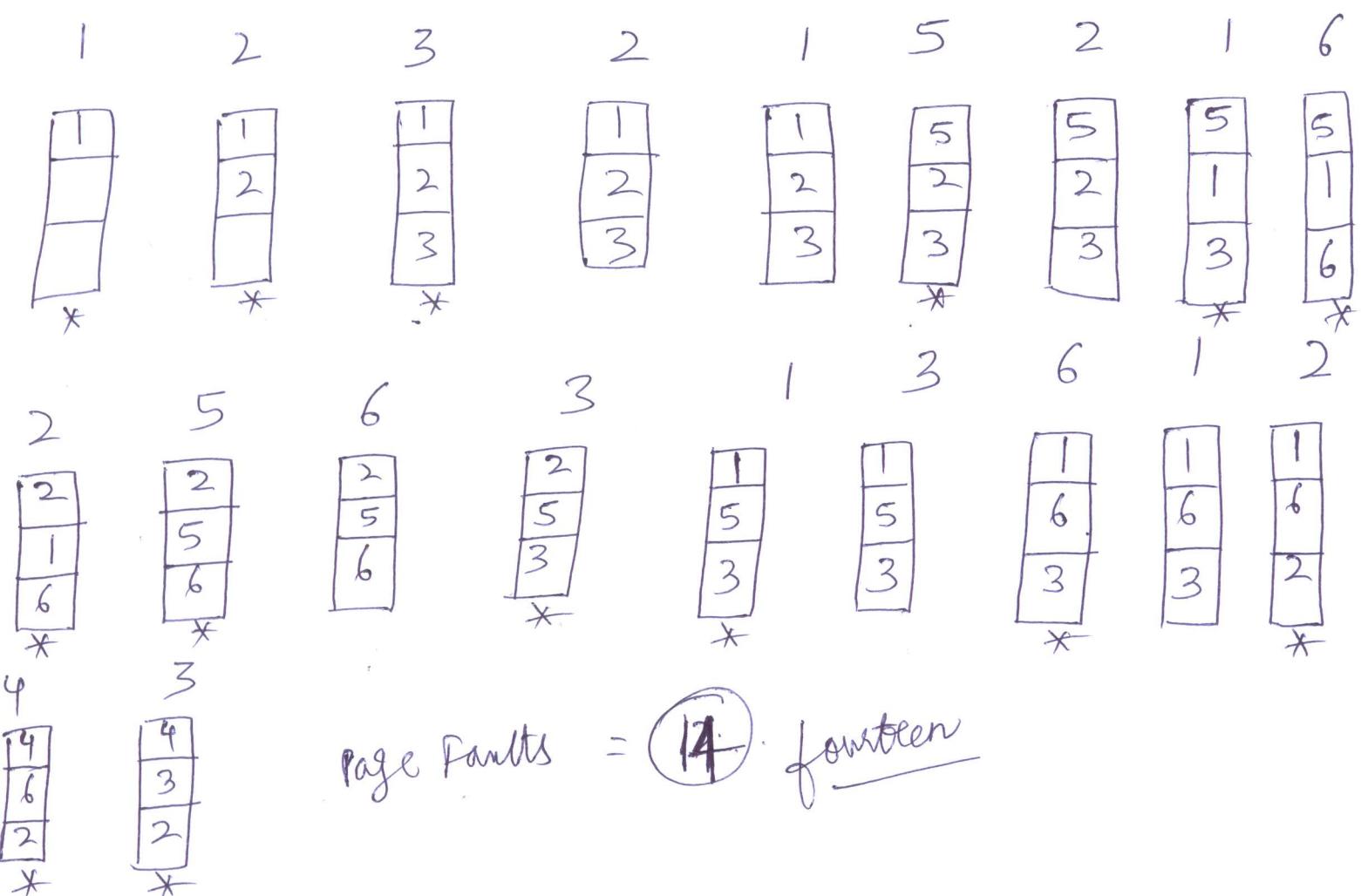
- It allows cycles.
- It is more flexible than other directories structure.

**Disadvantages:**

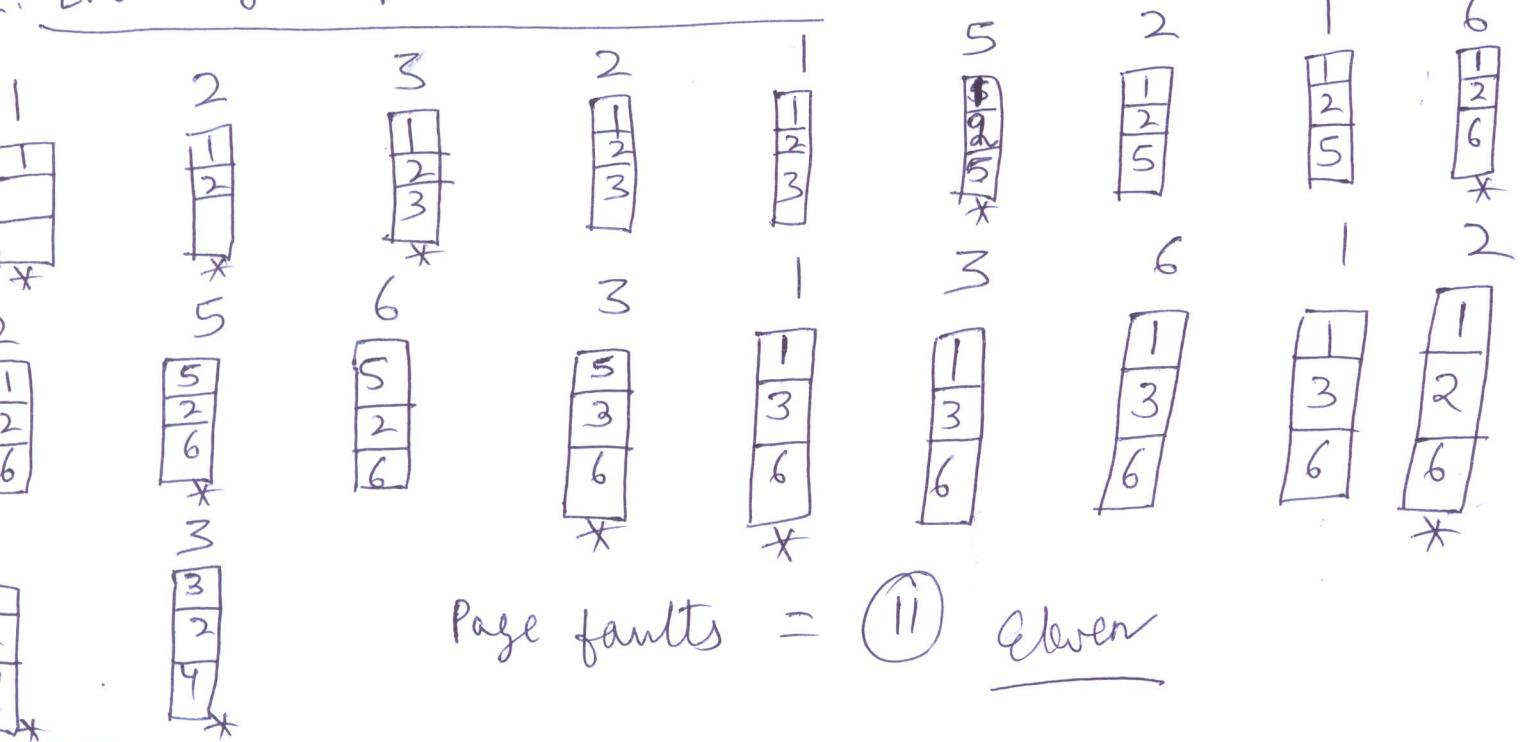
- It is more costly than others.
- It needs garbage collection.

1.

## FIFO Page Replacement Algorithm



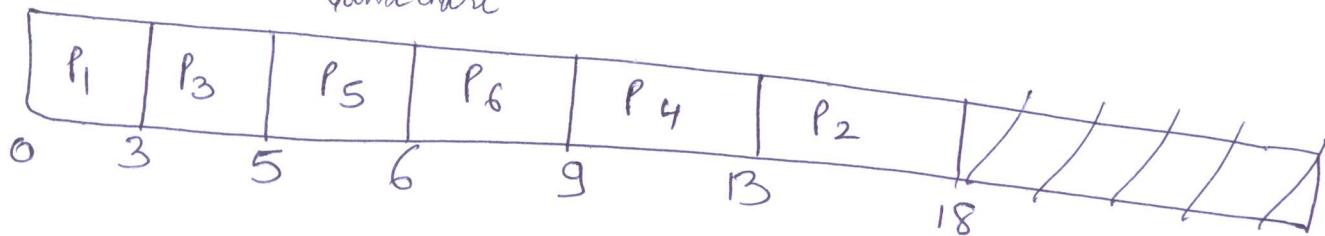
## LRU Page Replacement Algorithm:



Pid	AT	BT
P <sub>1</sub>	0	30
P <sub>2</sub>	1	50
P <sub>3</sub>	2	20
P <sub>4</sub>	3	40
P <sub>5</sub>	4	10
P <sub>6</sub>	5	30

Preemptive SJF

Gantt chart



$$\begin{array}{l}
 \text{WT for } P_1 \rightarrow (0-0) = 0 \\
 P_2 \rightarrow (13-1) = 12 \\
 P_3 \rightarrow (3-2) = 1 \\
 P_4 \rightarrow (9-3) = 6 \\
 P_5 \rightarrow (5-4) = 1 \\
 P_6 \rightarrow (6-5) = 1
 \end{array}
 \quad
 \begin{array}{l}
 \text{TAT for } P_1 \rightarrow 0+3 = 3 \\
 P_2 \rightarrow 12+5 = 17 \\
 P_3 \rightarrow 1+2 = 3 \\
 P_4 \rightarrow 6+4 = 10 \\
 P_5 \rightarrow 1+1 = 2 \\
 P_6 \rightarrow 1+3 = 4
 \end{array}$$

$$\text{Total WT} = 21$$

$$AWT = \frac{21}{6} = 3.5$$

$$\text{Total TAT} = 39$$

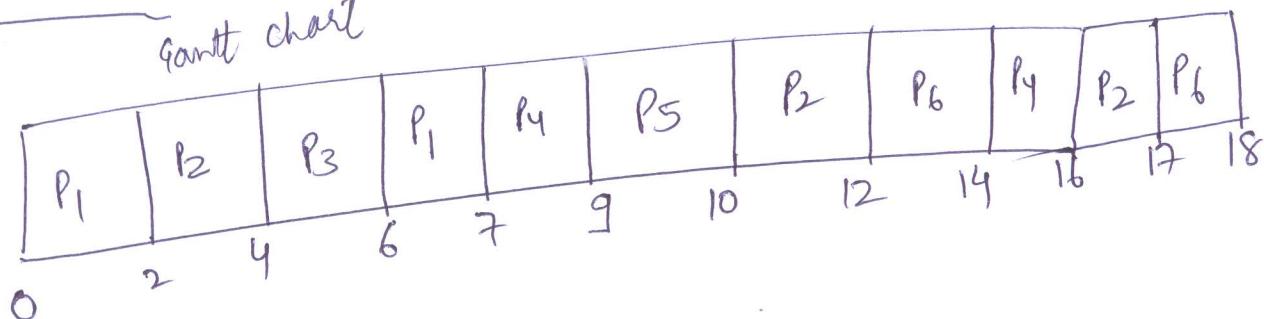
$$AWT = \frac{39}{6} = 6.5$$

Ans 2

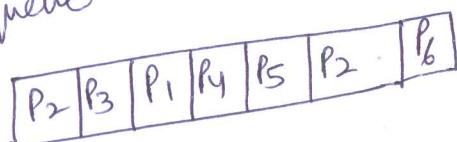
Pid	AT	BT
P <sub>1</sub>	0	3+0
P <sub>2</sub>	1	5 3+0
P <sub>3</sub>	2	20
P <sub>4</sub>	3	4 20
P <sub>5</sub>	4	+0
P <sub>6</sub>	5	3+0

Round Robin

Gantt chart



Ready queue



WT for P<sub>1</sub>  $\rightarrow (0-0) + (6-2) = 4$

P<sub>2</sub>  $\rightarrow (2-1) + (10-4) + (16-12) = 11$

P<sub>3</sub>  $\rightarrow (4-2) = 2$

P<sub>4</sub>  $\rightarrow (7-3) + (14-9) = 9$

P<sub>5</sub>  $\rightarrow (9-4) = 5$

P<sub>6</sub>  $\rightarrow (12-5) + (17-14) = 10$

TWT = 41

AWT =  $\frac{41}{6} = 6.83$

TAT for P<sub>1</sub> = 4+3 = 7  
 P<sub>2</sub> = 11+5 = 16  
 P<sub>3</sub> = 2+2 = 4  
 P<sub>4</sub> = 9+4 = 13  
 P<sub>5</sub> = 5+1 = 6  
 P<sub>6</sub> = 10+3 = 13

TOTAL TAT = 59

AWT =  $\frac{59}{6} = 9.83$

### **Ans3.**

#### Palm OS

Palm OS is a proprietary mobile operating system. Designed in 1996 for Palm Computing, Inc.'s new Pilot PDA, it has been implemented on a wide array of mobile devices, including smartphones, wrist watches, handheld gaming consoles, barcode readers and GPS devices.

Palm OS versions earlier than 5.0 run on Motorola/Freescale DragonBall processors. From version 5.0 onwards, Palm OS runs on ARM architecture-based processors.

The key features of the current Palm OS Garnet are:

- Simple, single-tasking environment to allow launching of full screen applications with a basic, common GUI set
- Monochrome or color screens with resolutions up to 480x320 pixel
- Handwriting recognition input system called Graffiti 2
- HotSync technology for data synchronization with desktop computers
- Sound playback and record capabilities
- Simple security model: Device can be locked by password, arbitrary application records can be made private
- TCP/IP network access
- Serial port/USB, infrared, Bluetooth and Wi-Fi connections
- Expansion memory card support

#### Symbian OS

- Design
  - Symbian features pre-emptive multitasking and memory protection
  - Symbian OS was created with three systems design principles in mind:
    - the integrity and security of user data is paramount
    - user time must not be wasted
    - all resources are scarce
  - Symbian uses a microkernel, has a request-and-callback approach to services, and maintains separation between user interface and engine.
  - The OS is optimised for low-power battery-based devices
  - Applications, and the OS itself, follow an object-oriented design: Model-view-controller (MVC).
  - Later OS iterations diluted this approach in response to market demands, notably with the introduction of a real-time kernel and a platform security model in versions 8 and 9.
  - There is a strong emphasis on conserving resources
- Features
  - User interface

Symbian has had a native graphics toolkit since its inception, known as AVKON (formerly known as Series 60). S60 was designed to be manipulated by a keyboard-like interface metaphor, such as the ~15-key augmented telephone keypad, or the mini-QWERTY keyboards. AVKON-based software is binary-compatible with Symbian versions up to and including Symbian^3.

- Browser

Symbian^3 and earlier have a built-in WebKit based browser. Symbian was the first mobile platform to make use of WebKit (in June 2005). Some older Symbian models have Opera Mobile as their default browser.

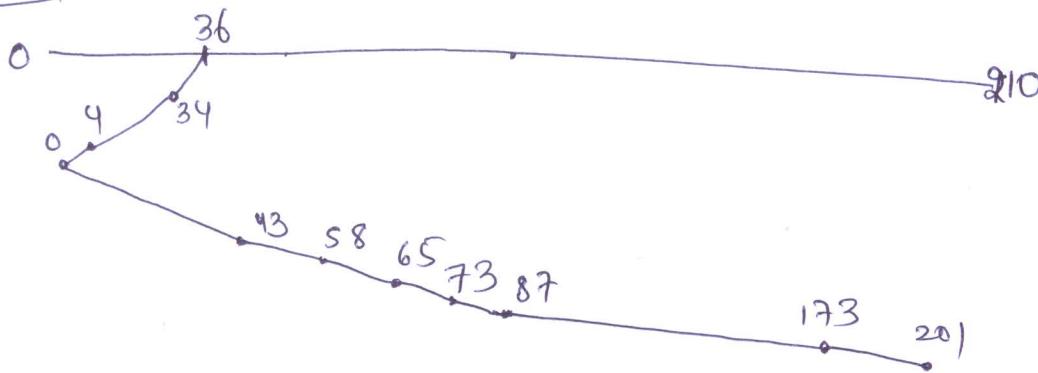
- Multiple language support

Symbian has strong localization support enabling manufacturers and 3rd party application developers to localize their Symbian based products in order to support global distribution. Current Symbian release (Symbian Belle) has support for 48 languages, which Nokia makes available on device in language packs (set of languages which cover the languages commonly spoken in the area where the device variant is intended to be sold). All language packs have in common English (or a locally relevant dialect of it).

Ans 4.

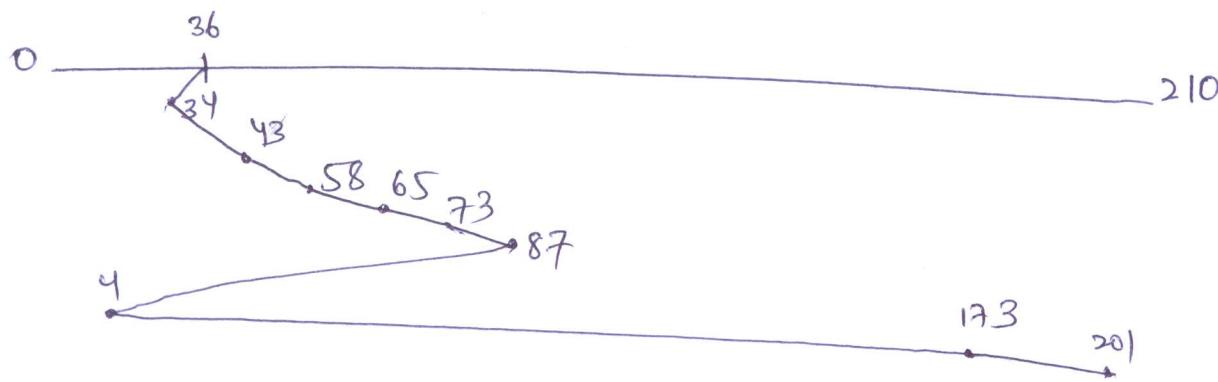
Sequence  $\rightarrow$  73, 87, 34, 43, 173, 65, 58, 4, 201

i) Scan



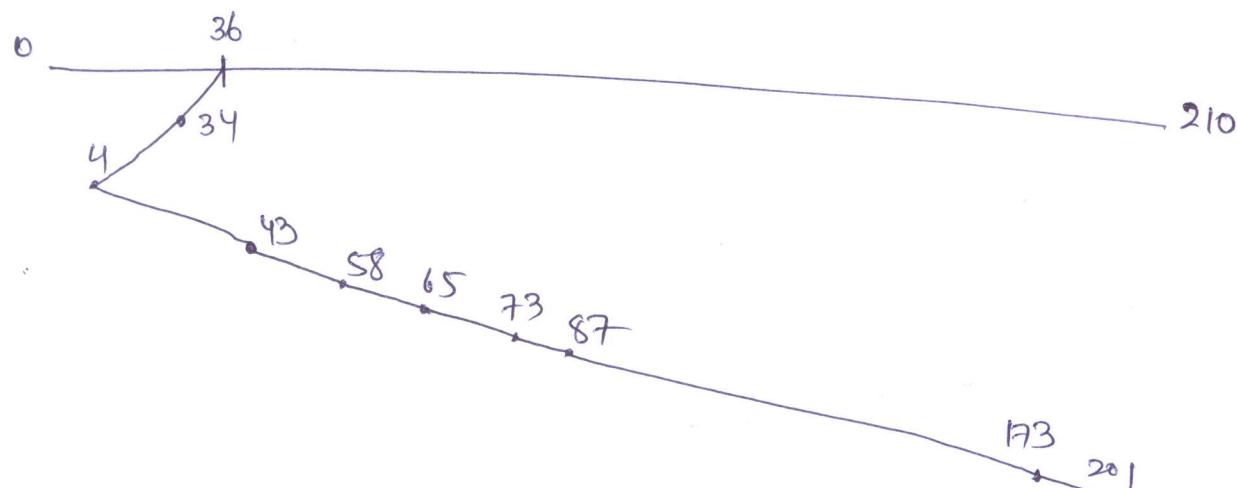
$$\begin{aligned}
 \text{seek time} &= (36-34) + (34-4) + (4-0) + (43-4) + (58-43) + (65-58) \\
 &\quad + (73-65) + (87-73) + (173-87) + (201-173) \\
 &= 237
 \end{aligned}$$

ii) SSTF



$$\begin{aligned}
 \text{seek time} &= (36-34) + (43-34) + (58-43) + (65-58) + (73-65) + (87-73) \\
 &\quad + (87-4) + (173-4) + (201-173) \\
 &= 335
 \end{aligned}$$

iii) Look



$$\begin{aligned}\text{Seek time} &= (36-34) + (34-4) + (43-4) + (58-43) + (65-58) \\ &\quad + (73-65) + (87-73) + (173-87) + (201-173) \\ &= 229.\end{aligned}$$

251

### **Ans5.**

Threads are smallest unit of execution and are typically spawned for a short-term benefit that is usually visualized as a serial task, but which doesn't have to be performed in a linear manner (such as performing a complex mathematical computation using parallelism, or initializing a large matrix), and then are absorbed when no longer required.

#### **Advantages**

- Responsiveness: If a thread gets a lot of cache misses, the other thread(s) can continue, taking advantage of the unused computing resources, which thus can lead to faster overall execution.
- Resource sharing: If a thread cannot use all the computing resources of the CPU (because instructions depend on each other's result), running another thread can avoid leaving these idle.
- Utilization of multiprocessor architectures. The benefits of multithreading can be greatly increased in a multiprocessor architecture, where threads may be running in parallel on different processors. A single threaded process can only run on one CPU, no matter how many are available.
- Economy: Allocating memory and resources for process creation is costly. Because threads share resources of the process to which they belong, it is more economical to create and context-switch threads.

#### **Disadvantages**

- Execution times of a single thread are not improved but can be degraded, even when only one thread is executing. This is due to slower frequencies and/or additional pipeline stages that are necessary to accommodate thread-switching hardware.
- Hardware support for multithreading is more visible to software, thus requiring more changes to both application programs and operating systems than multiprocessing.
- Thread scheduling is also a major problem in multithreading.