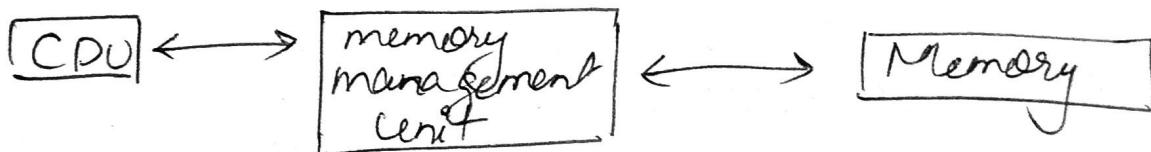


MEMORY MANAGEMENT

Address Binding

The process of mapping from one address space to another address space i.e. It is the mapping or translation of logical address into physical address. Logical address is address generated by CPU during execution whereas physical address refers to location in memory unit.



In address Binding done in 3 ways.

1. Compile time address Binding
2. Execution time / Dynamic address binding.
3. Load time address Binding.

Compile Time Address Binding

If the compiler is responsible for performing Address Binding then it is called compile time address binding.

Execution time / Dynamic Address Binding

This is done by the processor at the time of program execution.

Load Time Address Binding.

If it is not known at the compile time where the process will reside, then a relocatable address will generated the loader then translate the relocatable address to an absolute address. This type of address binding will be done by the operating system memory manager that is loader.

Logical & Physical Address in OS.

An address generated by CPU is commonly referred to as logical address. The set of all logical address generated by a program is called a logical address space. The logical address does not exist physically in the memory. An address seen by the memory unit that is the one loaded into the memory address register of the memory is commonly known as a physical address. The set of all physical address corresponding to their logical address is a physical address space. Physical address is a location in memory that can be accessed physically.

Dynamic Loading

It is necessary for the entire program & all data of a process to be in physical memory for the process to execute the size of a process has thus been limited to the size of physical memory. To obtain the better memory space utilisation we can use dynamic loading with dynamic loading a routine is not loaded until it is called. All routine are kept in the relocatable load format. The main program is loaded into memory & is executed when a routine needs to call another routine the calling routine first checks to see whether the other routine has been loaded. If it has not the relocatable linker loader is called to load the desire routine into memory & to update the program's address table to reflect this change. The advantage of dynamic loading is that a routine is loaded when it is needed. Dynamic loading does not require a special support from us. It is responsibility of the user to design their program to take advantage of such a method.

Dynamic Linking & Shared Libraries

Dynamically linked libraries are system libraries that are linked to user programs when the programs are run. Some OS supports static linking copies all libraries modules used in the program into the final executable file at the final step of the compilation while dynamic linking. The linking occurs at run time when both executable file & libraries are placed in the memory. Dynamic linking loads the routines when they are needed then the time & memory space will be same.

Shared libraries

Libraries that are linked dynamically. It allows OS code to be bounded into memory & used by any application software on the system without loading multiple copies into memory. All the applications on the system can use it without using more memory.

Loading & Linking

Bringing the program from secondary memory into main memory is called Loading. It loads the executable file obtained from the linking into main memory for execution. It in takes the executable module generated by linking. It allocates space to an executable module in main memory.

Establishing the linking b/w all the module of the program in order to continue the program execution is called linking. It generates the executable file of a program. It in takes the object module of a program generated by a assembler.

Swapping

It is a mechanism in which a process can be swapping temporarily out of main memory to disc & make the memory available to other process. It is also known as a technique for memory compaction.

Overlays

Memory is limited & the CPU can only have a limitation amount of space given to a process to

enable efficient CPU utilization, overlays come into a picture / point. It is a piece of code. It reduce time & memory requirement contiguous memory allocation.

Memory management technique are basic technique that are used in managing the memory in OS. This technique are basically classified into two category.

- 1) Contiguous
- 2) Non-contiguous.

Contiguous

It is basically a method in which a single contiguous section or part of memory is allocated to a process. Because of this all the available memory space resides at the same place together which means that the freely available memory partitions are non distributed in a random fashion here & there across all the memory space. It has two types:-

- 1) Fixed / Static partition scheme
- 2) Variable / Dynamic "

Fixed / Static partition scheme

In this scheme memory is divided into fixed no. of partitions at the time of system configuration. It's self fixed means no. of partitions are fixed in the memory not same sized every partition will be accommodate only one process. So one partition contains one process. e.g.: - P_1, P_2, P_3 .

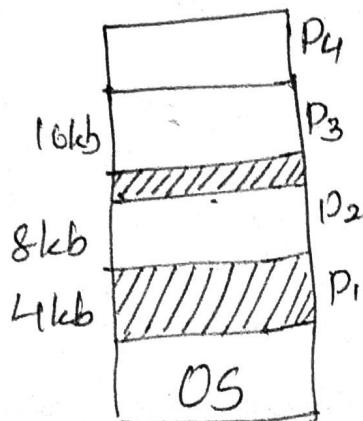
Degree of multiprogramming is restricted by no. of partition in the memory. Every partition is associated with the limit register.

Advantages

- Easy to Implement :- algorithms needed to implement fixed partition are easy to implement.
- Little OS overhead :- Processing of fixed partition in requires lesser and indirect computation power.

Disadvantages

Internal fragmentation :- when a process is allocated to a memory block & if the process is smaller than the amount of memory request a free space is created in the given memory block due to this the memory block is unused. Because the partition can accommodate only one process which causes internal fragmentation.



- To overcome this problem of internal fragmentation instead of fixed partition scheme, variable partition scheme is used.
- Limitation on degree of multiprogramming : no of processes greater than the no. of partition in RAM is invalid in fixed partitioning.

2. Variable / Dynamic Partitioning Scheme

To overcome the limitations of fixed partitioning

technique. This technique is called MVR. It is used. It is generalization of the fixed partitioning technique in which the partition can vary in no. & size. In this Technique the amount of memory allocated is exactly the amount of memory process requires to implement this, the table contained by OS stores both the starting & ending address of each process. In this partitioning space in main memory is allocated strictly according to the need of process. Hence there is no case of internal fragmentation. It also provides the concept of compaction.

Advantages:

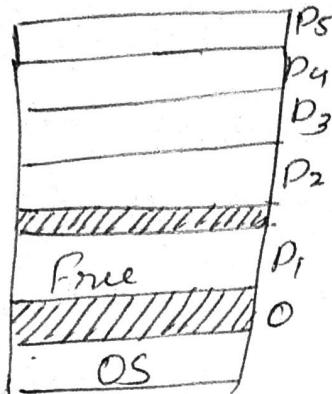
- No internal fragmentation
- No restriction on degree of multiprogramming
- No limitation of the size of the process

Disadvantages:

- Difficult Implementation
- External Fragmentation

It occurs when there is a sufficient amount of space in the memory to satisfy the memory request of a process but the process's memory request cannot be satisfied as the memory

available in a non contiguous manner.



Techniques to avoid external fragmentations are

1. Compaction.
2. Partition selection algorithms (first, best, worst)
3. Paging
4. Segmentation.

Compaction

Moving all the process or towards the top or bottom to wait free available memory to a single continuous place is called compaction.

Partition Selection Algorithm

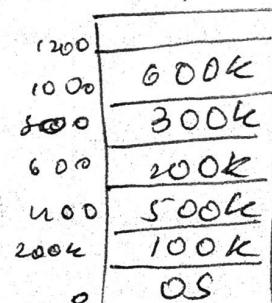
- First Fit : In this algorithm the OS scans the free storage list and allocates the first hole that is large enough to accommodate that

process this is fast because search is little as compared to other algorithms.

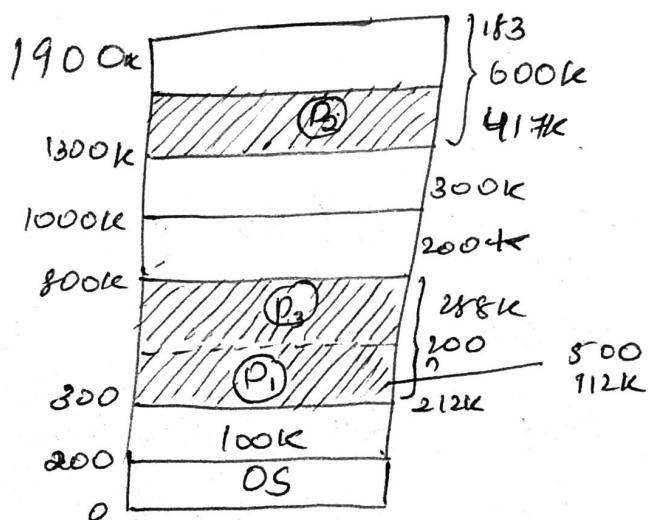
- Best fit: In this, the OS scans the free storage list & allocates the smallest hole whose size is larger than or equal to the size of the process. It is slower than first fit algorithm as it has to search the entire list every time.
- Worst fit: In this, the OS scans the free storage list & allocates the largest hole to the process.

Q) Given memory partition of 100k, 500k, 200k, 300k & 600k in order. How would each of the this 3 fit algorithm. Place the process of 212k, 417k, 112k, & 426k which algorithm makes the most efficient of memory show the diagram in each case.

ans. Assuming that the OS resides lower part of main memory & occupies 200k



According to 1st fit algorithm the 1st process of size 212K is placed in 500K partition resulting in a hole of size 288K. The next process of size 417K is placed in 600K partition resulting in a hole of size 183K. The next process 112K is placed in 288K partition resulting in a hole of size 176K. Since now there is no partition big enough to accommodate the size of last 426K. The process has to wait.



According to best fit algorithm - The first process of size 212K is placed in 300K partition resulting in hole of size 88K the next process to be placed (417K) is placed in 500K resulting - - . 83K

went \rightarrow 192k \rightarrow placed \rightarrow 200m partition \rightarrow 88k

last process → 426k → placed → 600k partition → 174k
balance

According to worst fit algorithm - The

1st process \rightarrow 600K partition \rightarrow hole of

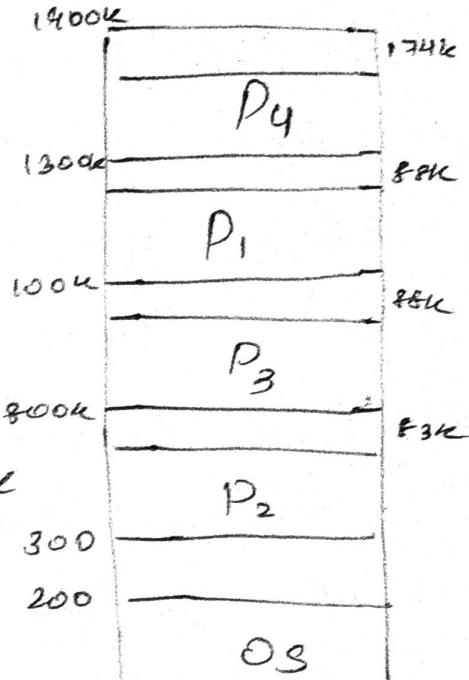
size 388K \rightarrow next process \rightarrow 417K

\rightarrow hole of size 276K - Since now

there is no partition big enough to

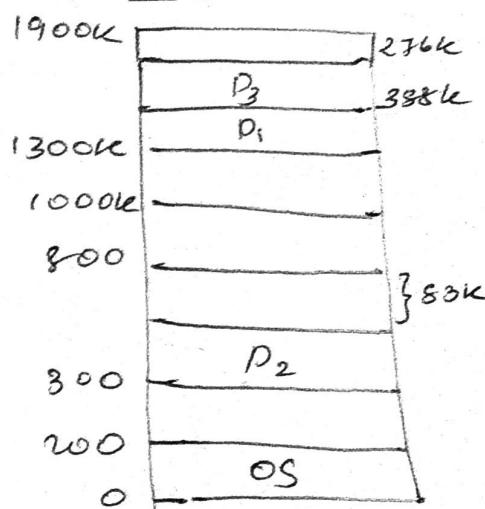
accommodate the last process of size

426K the process has to wait.



Advantages of Contiguous Allocation

- Faster in execution
- It allocates consecutive blocks of memory to process.
- It is easier for the OS to control.
- Overhead is minimum.
- It includes single & multi-partitions allocation.

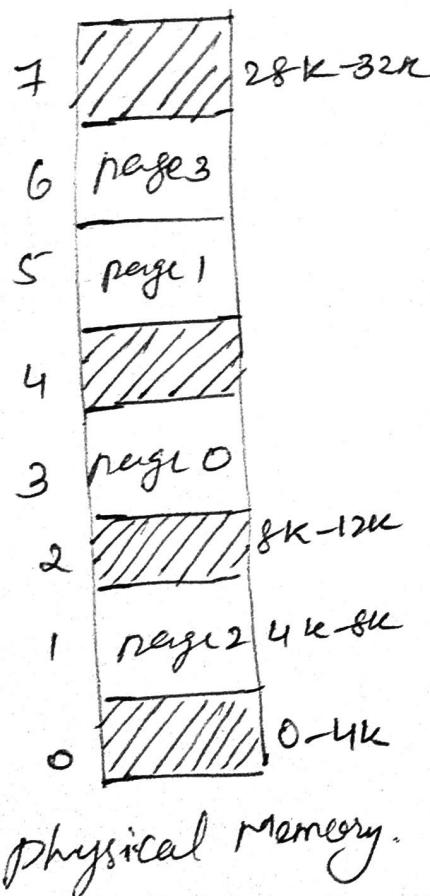


3) Paging

In this the physical memory is divided into fixed sized blocks called page frames & logical m/m divided into fixed sized blocks called pages which are of same size as that of page frames.

When a process is to be executed. Its pages can be loaded into any unallocated frames from the disk.

• address translation is performed used mapping table called page table. The OS maintains a page table for each process to keep track of which page frame is allocated to which page when CPU generates a logical address. That address is send to memory manager. The manager uses the page no. to find the corresponding page frame number in the page. That page frame no. is attached to the higher order end of the page offset to form the physical address that is send to the memory.



page no. Frame no.

0	3
1	5
2	1
3	6

Page Table

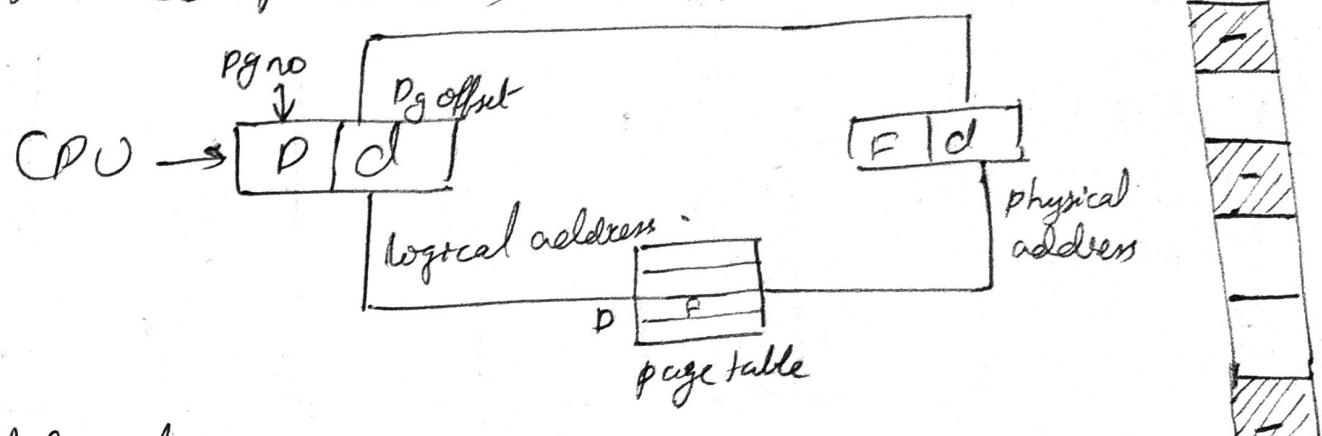
page 3
page 2
page 1
page 0

logical mem

Physical Memory.

— OS decides how much will the size of each frame that can be supported in flat memory only, condition

has to be satisfied is frame size & page size must be equal. The hardware support for paging is given below. Every address generated by the CPU is divided into 2 parts - at page no (P) & page offset (displacement)



Advantages

- Easy to use MMU
- No need for external fragmentation
- Swapping is easy b/w equal size of pages & page frames.

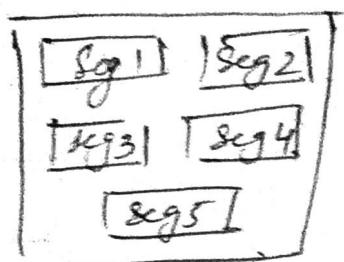
Limitations

- Page tables consume additional m/m.
- May cause internal fragmentation.

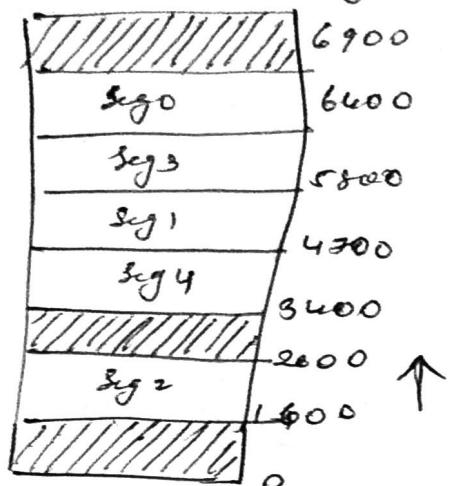
4) Segmentation

It is a m/m technique in which the m/m is divided into variable size part each part is known as segment which can be allocated to a

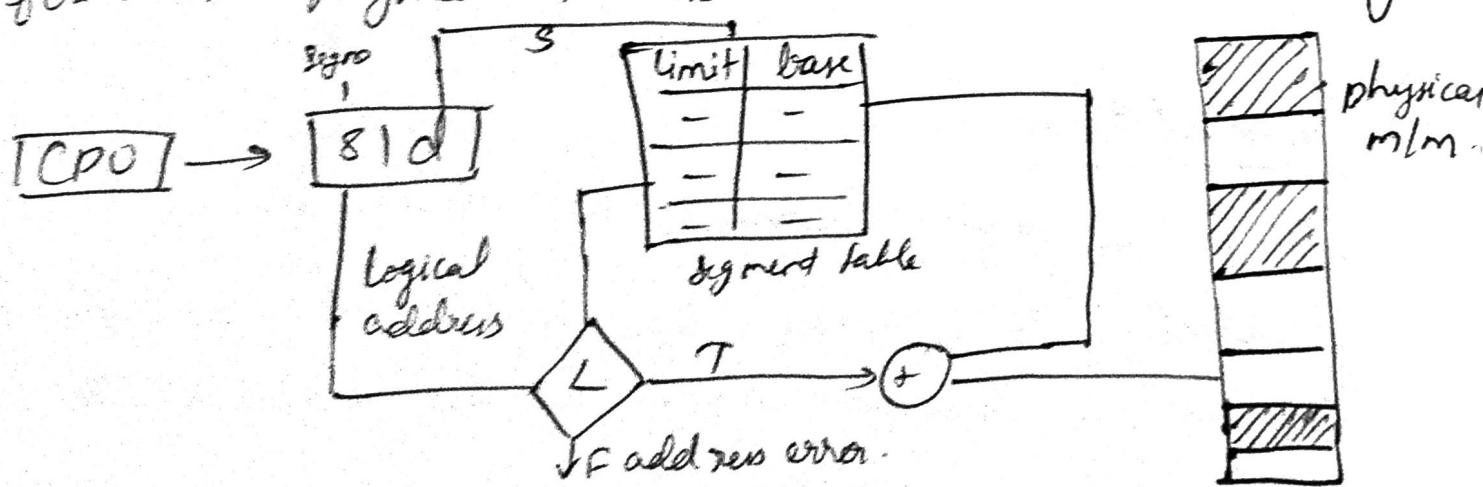
process. This method works similar to paging. Only difference b/w the 2 is that segments are of variable length whereas pages are of fixed size. The details about each segments are stored in a table called segment table, is maintained by OS.



	size	length
0	6400	500
1	4300	110
2	1600	1000
3	5800	600
4	3400	1200



When CPU generates a logical address the address is send to MMU. The MMU uses the segment no. of logical address as an index to the segment table. The offset is compared with the segment limit & if it is greater, invalid address error is generated. On the offset is added to the segment base to form the physical address ie send to the memory.



Virtual Memory

It is a technique that allows executing a program in partially in memory. In this technique the OS loads only those parts of a program in memory that are currently needed for the execution of the process. The rest part is kept on the disc & is loaded into the memory only when needed.

In ~~real~~ virtual memory systems the logical address is referred to as virtual address & logical address space is referred to as ^{virtual} address space.

Demand paging

A page is loaded into the memory only when it is needed ~~into the~~ ^{during} program execution. Pages that are never accessed are never loaded into the m/m. Whenever a process is to be executed, an area on secondary storage device is allocated to it on which its pages are copied the area is known as swap space of the process. During the execution of the process whenever a page required it is loaded into the main m/m from the swap space. If a process is to be removed from main memory it is

returns back into the swap space if it been modified.

- If process request for a ~~space~~ page the virtual address is send to MMU. The MMU checks the ~~wanted~~ valid bit in the page table. If the valid bit is one, it is accessed as a pageing. Otherwise the MMU raises an interrupt called page fault.

A demand paging system combines the features of paging with swapping. It reduces the swap time since only the required pages are swapped in instead of swapping the whole process. It increases the degree of multiprogramming by reducing the amount of physical m/m required for a process. It minimizes the initial disc overhead.

Page Replacement

When page fault occurs, page fault routine occurs a free page frame in m/m & allocates it to the process. However there is a possibility that the m/m is full. In that case the OS has to evict a page from the m/m to make the space for the desired page to be

swapped in. The page to be evicted will be returned to the disc depending on whether it has been modified or not. If the page has been modified while in the m/m it is returned to the disc otherwise no rewrite is needed. When a page is first loaded into the m/m the modified bit is cleared. It is set by the hardware when any word or byte in the page is written into. At the time of page replacement if modified bit for a selected page is cleared, it implies that the page has not been modified & since it was loaded into the m/m. The page frame is returned back to the swap space only if modified bit is set. The replacement algorithm determining the no of page faults using a reference string. A reference string is an ordered list of m/m references made by a process. It can be generated by a random no. generator or recording the actual m/m references made by an executing program.

FIFO Page Replacement

- It is simplest form algorithm.
 - As the name suggest, the first page loaded into the m/m, # is the first to be replaced.
- e.g:- reference string.

[5 0 | 5 | 3 | 3 | 5 | 2 | 5 | 0 | 1 | 0 | 7 | 3 |]

frame	1	2	3	1	2	3	1	2	3	1	2	3
3												
2	0	0	0	0	0	5	5	5	5	7	7	3
1	5	5	5	5	8	5	2	2	1	1	1	1

M M H M H M M H M M M

$$\text{No. of faults} = \frac{9}{12}$$

$$\text{No. of hits} = \frac{3}{12}$$

Initially all the 3 frames are empty. The first 2 references made to page 5 & 0 & cause page fault. As the result they are swapped in m/m. Their 3rd reference made to page 5, does not cause page fault as it is already in m/m. i.e. page hit (H).

The next reference made to page 3 causes a page fault & that page is brought to m/m.

The reference to page 10 causes a page fault which results in the replacement of page 5 as it is the oldest page. This process continues until all the pages of reference strings are accessed. This algorithm is easier to implement as compared to all other replacement algorithms. It is rarely used as it is not very efficient.

Reference String

4	7	6	1	7	6	1	2	7	2	1
---	---	---	---	---	---	---	---	---	---	---

3 2 1 0

frame	3	2	1	0	1	2	3	4	5	6
3										
2	6	6	6	6	6	6	8	7	4	
1	7	7	7	7	7	#	2	2	2	
0	4	4	4	1	1	1	1	1	1	2

$$\text{No. of faults} = \frac{6}{10}$$

$$\text{No. of hits} = \frac{4}{10}$$

Belady's Anomaly

While using the FIFO replacement algorithm

the no. of page fault increases by increasing the no. of frames. This phenomena is called Belady's anomaly.

e.g:-)

4	7	6	1	7	6	1	2	4	2
---	---	---	---	---	---	---	---	---	---

frame	4	3	2	1
4		6	6	6
3		6	6	6
2	7	7	7	7
1	4	4	4	4
	M	M	M	H
				H
				H
				H

$$\text{No. of faults} = \frac{M}{10} = \frac{5}{10} = 2$$

$$\text{No. of hits} = \frac{H}{10} = \frac{5}{10} = 2$$

2)

5	0	1	5	3	5	1	2	5	1	0	1	0	7	3
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

		2	*	2	2	2	2	2	2	2	2	2	2	2
		3	3	3	3	3	3	3	3	3	3	3	3	3
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
5	5	5	5	5	5	5	8	1	1	1	1	1	1	1

M M H H H M H M M M H M H

$$\text{No. of faults} = \frac{M}{D}$$

$$\text{No. of hits} = \frac{H}{D}$$

Optimal (OPT) Page Replacement

- It is the best possible page replacement algorithm
- Pages are replaced which would be used for the longest duration of time for future i.e. the pages in the WFM which are going to be referred farthest in the future are replaced.

Eg:- 2 3 4 2 1 3 7 5 4 3

		4	4	4	4	4	4	4	4	4	4	4	4	4
		3	3	3	3	3	3	3	3	3	3	3	3	3
2	2	2	2	1	X	X	5	5	5	5	5	5	5	5

M M M H M H M M M M M M M

$$\text{No. of faults} = \frac{6}{10} = \frac{3}{5}$$

$$\text{No. of hits} = \frac{4}{10} = \frac{2}{5}$$

Advantages

- Easy to implement.
- It causes the lowest no. of page faults as compared to other algorithms.
- Highly efficient.
- Simple data structures are used.
- Disadvantages

- Time consuming.
- Requires future knowledge of the program.

6 7 8 9 6 7 1 6 7 8 9 1 7 9 6

1	2	9	9	9	1	1	1	1	1	1	1	1	1	1
3	3	3	7	7	7	7	7	7	7	7	7	7	7	7
6	6	6	6	6	6	6	6	8	9	9	9	9	6	
H	H	H	H	H	H	H	H	M	M	H	H	H	H	H

$$\text{No. of faults} = \frac{8}{10}$$

$$\text{No. of hits} = \frac{2}{10}$$

Least Recently Used (LRU)

- It is an approximation to the optimal algorithm.
- In this algorithm page will be replaced which is least recently used.
- This algorithm works on previous data, the page which is used the earliest is replaced or which appears the earliest in the sequence is replaced.
- LRU keeps track of page usage over a short period of time. It works on the idea that the pages that have been most used in the past are most likely to be used in the future too.

e.g:- 1 2 3 4 5 1 3 1 6 3 2 3

4			4	4	4	4	4	6	6	6	6
3		3	3	3	3	3	3	3	3	3	3
2	2	2	2	1	1	1	1	1	1	1	1
1	1	1	1	5	5	5	5	5	8	2	2

eg:- 2 0 1 2 0 3 0 4 2 3 0 3 2

3		1	1	1								
2	0	0	0	0								
1	7	7	7	2	2							
	M	M	M	M	M	14						

?

Advantages

- Efficient
- Less page faults as compared to FIFO
- It doesn't suffer from Lru's anomaly

Disadvantages

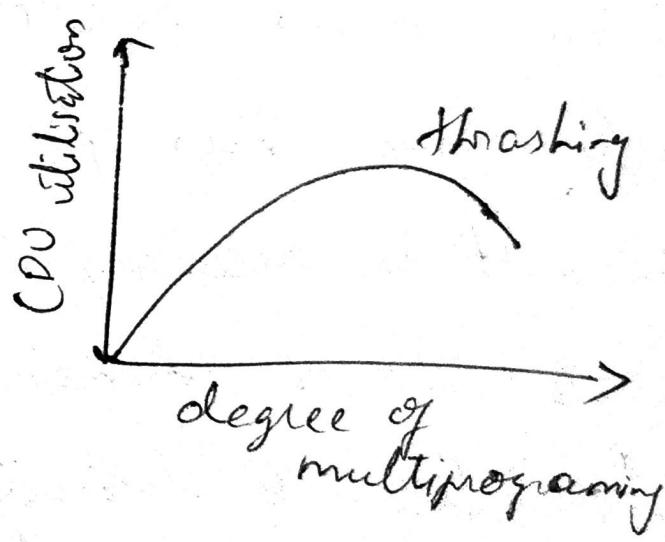
- Complex implementation
- Expensive
- Time consuming
- Requires hardware support

Thrashing

It is when the page fault & swapping happens very frequently at higher ratio.

then the OS has to spend more time swapping these pages. This statement in the OS is known as thrashing.

Consequently the process will quickly fault again & again, replacing pages that it must bring back in immediately. This high paging activity by a process is called thrashing. The system can detect thrashing by evaluating CPU utilization against the degree of multiprogramming. Increase the degree of multiprogramming the CPU utilization increases.



Locality

Thrashing can be prevented if each process is allocated as much memory as it requires but

The issue arises how the OS knows the m/m requirement of a process. The solⁿs of this problem is influenced by 2 factors. If a process is allocated more no. of frames than it requires, less page fault would occur. The process performance would be good. However degree of multiprogramming would be low. As a result CPU utilization & system performance would be poor. Under commitment of m/m for a process causes high page fault rate which would result in poor process performance. Thus for better system performance, it is necessary to allocate appropriate no. of frames to each process. Locality is defined as the set of pages that are actively used together.

Protection Security & OS

- Protection & security requires that computer to requires to resources such as CPU, software m/m etc have protected.
- Security deals with the threats to information caused by the outsiders whereas protection deals with the threats caused by other user.
- This can be done by ensuring integrity, confidentiality & availability in the OS. The system must be protect against unauthorised access, viruses etc.

Security

The security system covers the safety of their system resources across malignant alteration, illegal actions & disparity or inconsistency.

It gives a mechanism authentication & encryption to analyse. The user to permit for using the system.

Protection

It deals with the access to the systems resources. It determines that what files can be accessed or

permitted by a special user. The protection of the system should confirm approval of the process

Due to this then licensed users & processes will care for the CPU, m/m & alternative sources.

Threats to Protection & Security

A Threat is a programs that is malicious in nature ~~& deal~~ leads to the harmful effect of the system. Some of the common threats that occur in a system are 1) Virus - They are generally small snippets of code embedded in a system. They are very dangerous & can corrupt files, destroy data crash systems etc. They can also spread by replicating themselves as required. ~~then~~

2) Trojan horse

It can secretly access login details of a system. It enters into a computer system through an email or free programs that have been downloaded from the internet.

3) Spyware

They are the small programs that installs themselves on computers to gather data secretly about the computer, user without his/her consent & report the collected data to interested users or parties.

4) Malware

They can destroy a system by using its resources to extreme levels. It can generate multiple copies which claim all the resources & don't allow any other process to access them. It can shut down ^{a whole} the all system network.

5) trap door

It is a security breach that may be present in a system without the knowledge of the

systems -

6) Hackers & Phising

Hackers are the programmers break into others computer system in order to steal, damage or change the information as they want. While phising is a form of threat that attempt to steal the sensitive data. It with the help of fraudulent emails & messages.

Need of Protection in OS

There may be security risks like unauthorized writing, reading, modification or preventing the system from working effectively for authorized users. It helps to ensure data security, process security & program security against unauthorized user access. Its purpose is to ensure that only the system policies access programs & resources.

Authentication & Authorisation

These are the 2 basic security terms.

Authentication : it is a process of identifying someone's identity by assuring that the person is the same as what ^{he} is claiming to be.

It is used for both server & client.

User authentication can be based user knowledge (such as username, password), user possession & user attribute. It does not ensure tasks under a process one person can do, what files can view, read, update. It mostly identifies who the person or system is actually is. The different method that may provide protection & security for different computer systems are password based authentication.

a) Username or password

Each user has a distinct username & password & they need to enter correctly before they can access the system.

b) User key or Card

The users need to put punch a card into card slot or use the individual key or keypad on the keypad to access a system.

c) User attribute identification.

Different user attribute identifications that can be used as fingerprint Iris design or retina. These are unique for each user & are compared with the existing data in the db. User can only access the system if there is a match.

2) One Time Password (OTP)

These passwords provide a lot of security for authentication purposes. A OTP can be generated exclusively for a login everytime a user wants to enter the system. It cannot be used more than ones. The various

ways a OTP can be implemented are

a) Random numbers

The system can ask for numbers that corresponds to alphabet that are prearranged. This combination can be changed each time a login is required.

b) Secret key.

A hardware device can create a secret key related to the userid for login. This key can change.

3) Encrypted passwords

It is also a technique for authenticating access. Encrypted data is passed via the network which transfers and checks passwords, allowing data to pass without interruption.

4) Cryptography.

It aids in the data secure transmission. It introduces the concept of a key to protecting the data. The key is crucial in

The situation - When a user sends data, he encodes using data a computer that has the key , & the receiver must decode the data with the same key .

Authorisation

It is the process of granting someone to do something . It means a user can permit entry into the network but not be authorized to access a resource .

- It requires a set of rules that dictate the resources to which a user will have access
- It is not always necessary to access information over the network .

Protection Mechanism

- Protection domain :- a system consists of a set of objects that may be accessed by the processes An object can be hardware object or software object such as file , program , disk

each object is referred to a unique name & is accessible by the processes using some predefined operations

A domain is a collection of access rights where each access is a pair of (Object name, rights set). The object name is the name of object & right set is the set of operations that a process is permitted to perform on the object name.

Access control list-

It is an alternative method of recording access rights in a system. A list is associated with each object such as file that stores user names which can access the object & type of access allow to each user. When a user tries to access an object, the ACL is searched for that particular object. If that user is listed for the requested access, the access is allowed otherwise the user is denied access to the user file.

• ACL is classifying the users of the system into the following 3 categories.

- 1) Owner - the user who created the file. The Admin of the file has full access to a file & can perform all file operations.
- 2) Group - a set of users who need similar access permission for sharing the file is a group. A group user can read, write a file but can't execute or delete a file
- 3) Universe - all the other users in the system form the universe. The members of the universe can only read a file & not allowed to perform any other operations.

Mobile OS

It is an OS that runs only on mobile devices. It allows different the user to own other different app's on its mobile tablet etc.

They are a mixture of computer OS with

some additional features of the mobile. They are comparatively light & simple.

Types of Mobile OS

The android OS is most common OS among the mobile OS. It is an open source & free OS.

Google is the developer of android.

Apple IOS

After android it is one of the most popular OS. It is designed to run devices such as iphone, ipad etc.

It has many strong security features. Moreover like the android devices having play store, apple IOS contain appstore.

Symbian

It is a mobile OS that provides high level integration with communication. It is based on the java language.

It was developed by symbian ltd in 1998.

Nokia was the first company to release OS on its mobile phone.

Palm OS

It is also a mobile OS was developed by Palm Ltd. It was introduced in 1996. It is also known as pocket OS.

Web OS

It is a mobile OS, that was developed by Palm. Based on the Linux Kernel.

Up OS was the OS in its mobile & handhelds.

BlackBerry

Developed by RIM (Research in Motion).

It is generally designed for BlackBerry handheld devices.

Bada OS

Samsung is the launcher of this OS in 2010. It includes features like 3D graphics, apps installed.

Windows Mobile OS

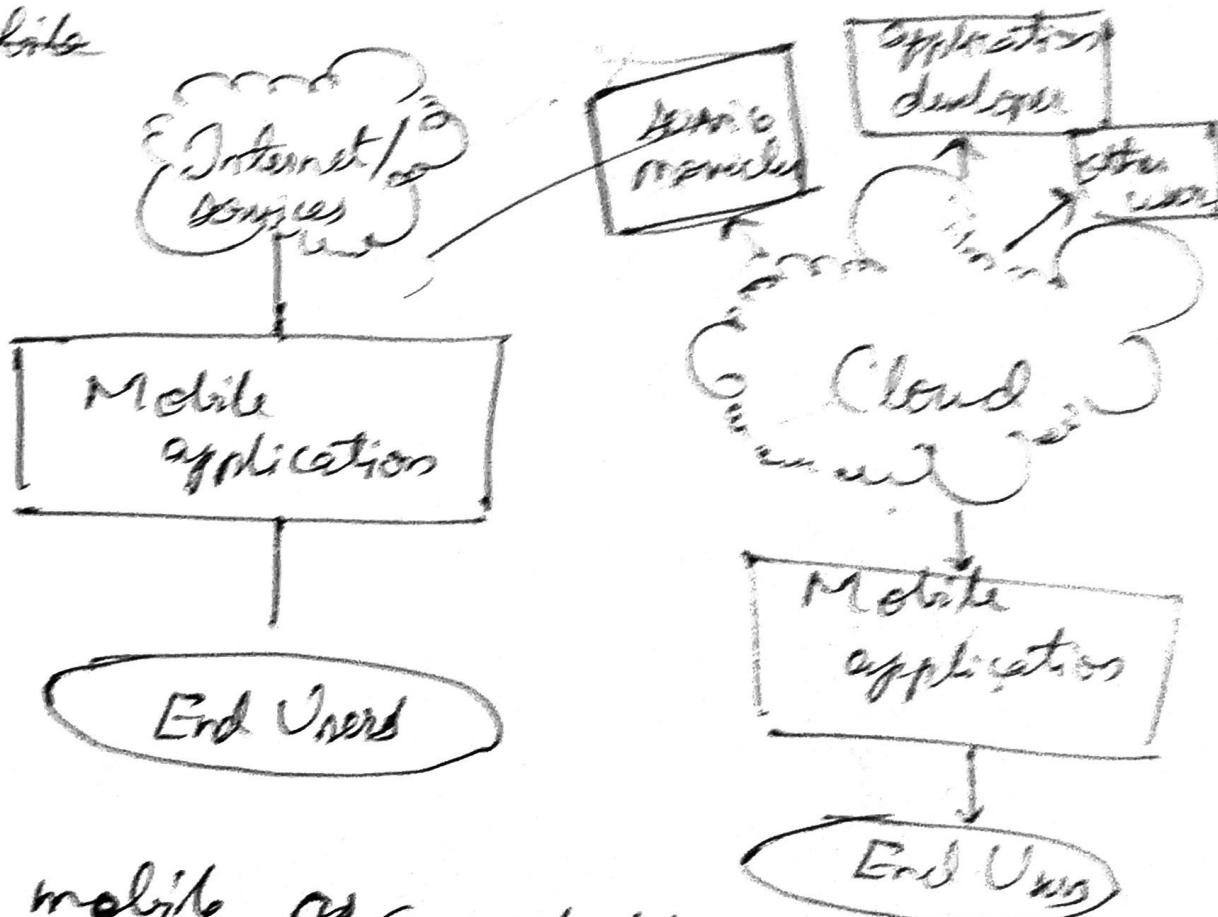
Developed by Microsoft, designed for pocket PCs & smartphones.

Features of M0s

- Easy to use
- Good app store
- Good battery life
- User friendly
- Data usage & organization.

Architecture of M0s

- Mobile phones are the most popular device
- Every website



Every mobile as a platform to run the other services like being easy for the users to use the services such as voice call, internet, sms etc.