unacademy

CS & IT Engineering

Operating Systems
Memory Management

Lecture Number- 25

By- Dr. Khaleel Khan Sir

unacademy

Memory Management

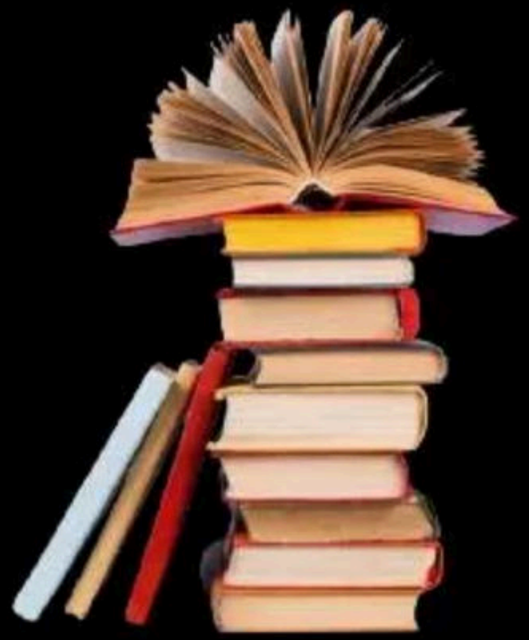
Comprehensive Course on Operating System

Khaleel Ur Rahman Khan • Lesson 26 • May 11, 2024

unacademy

Topics

to be covered



1. Addressing Vs Capacity ✓
2. Loading Vs Linking
3. Address Binding

unacademy

11. Loading vs Linking

Loading: refers to bringing the program (.exe) from Secondary Storage (disk) to Memory (RAM); It is done by Loader;

- Static (Loading bf Run Time)
- Dynamic

10KB
```
main()
{
  ...
  if (cond.)
    f();
  ...
}
```

5KB
```
f()
{
  ...
  g();
  ...
}
```

15KB
```
g()
{
  ...
  h()
  ...
}
```

2KB
```
h()
{
  ...
  &Gmf
  ...
}
```
12KB

54KB

Drawback of Static Loading : May Lead to ineffective utilization of Memory ; (Faster)

2) Dynamic Loading : Loading on demand @ Run Time ;

Adv: Space efficiency

Drawback : Prog. exec. Time increases (Time inefficiency)

2) Linking : The process of resolving external references, like variables, functions, Procedures, other objects of the Program;

→ Proto

```
#include <stdio.h>
extern int x; ✓ ext. ref
main()
{
    :
    f();      Addrs
    : BSA     ↓ ;
}
```

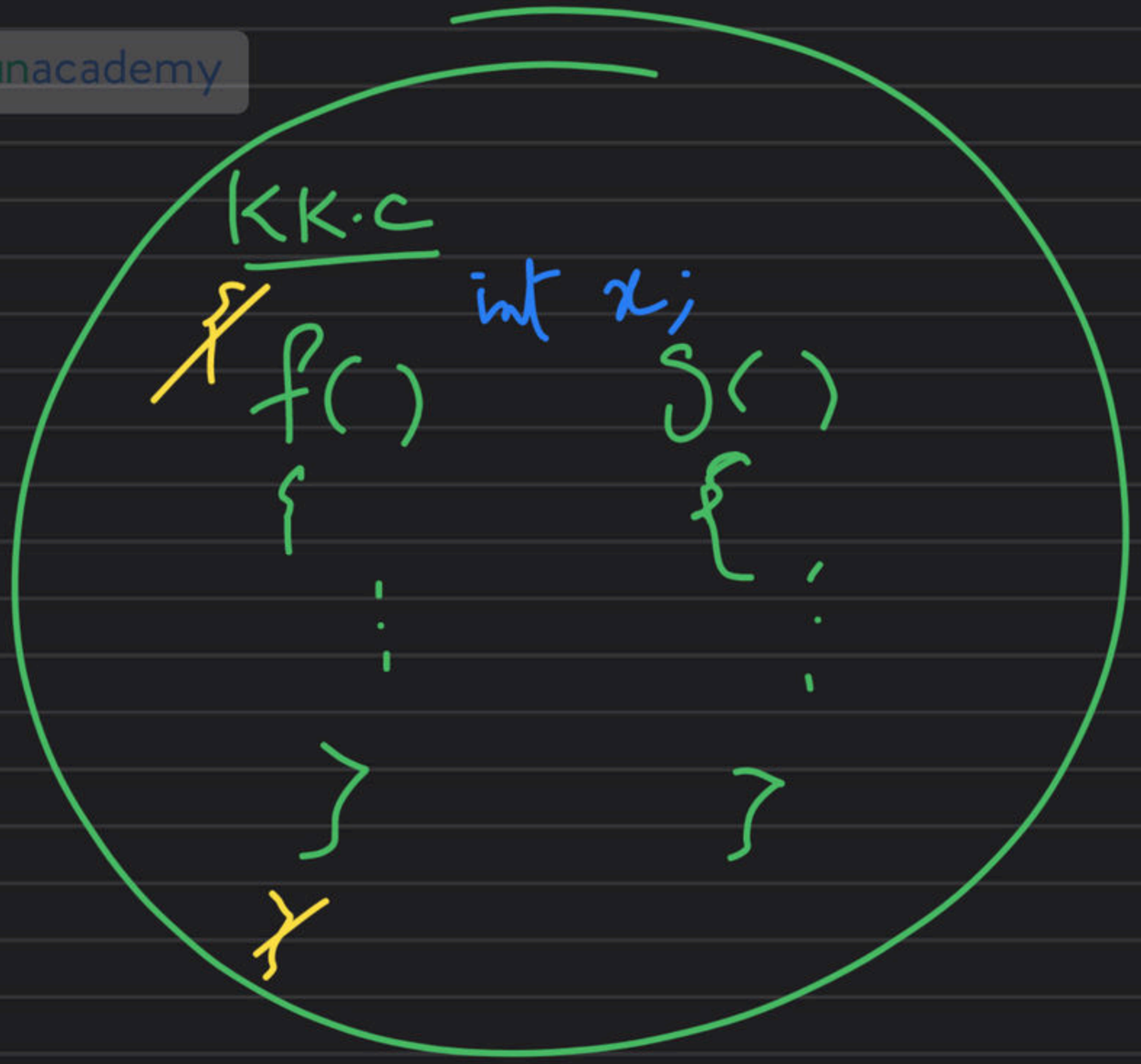
↓ Branch & Save Address

```
f()
{
    :
    BSA —
    g()
    :
}
```

```
g()
{
    :
    scanf(); BSA —
    :
}
```

.lib

linker
↓
(Linkage editor)



⟹ First Line

$ cc KK.c -o KK

KK (obj)

↓

May have unresolved references.

Linking

Static

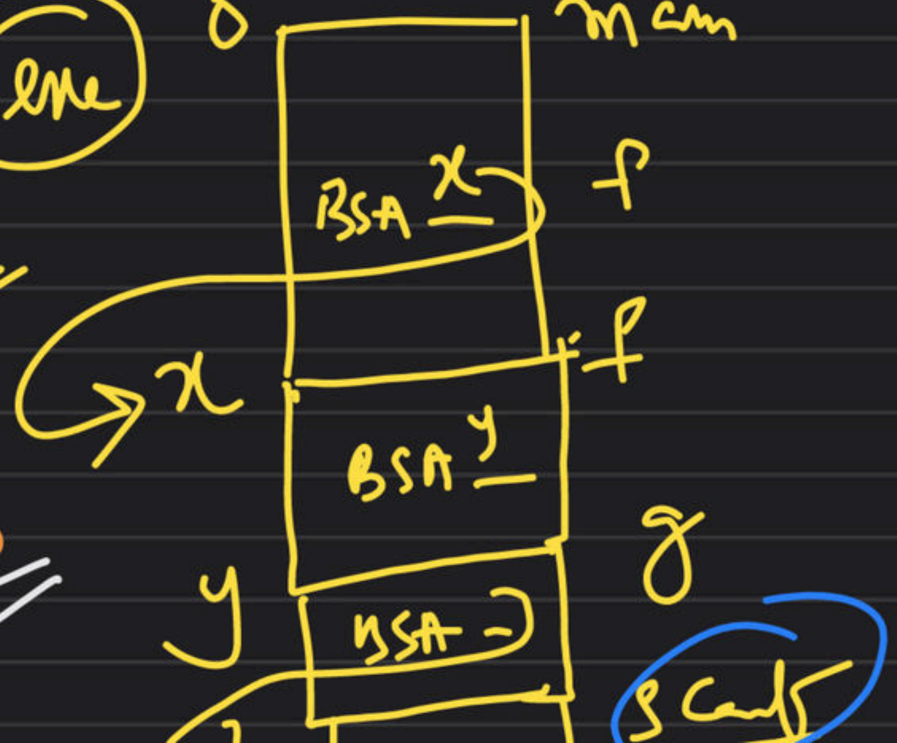
- Space inefficiency
- Time efficiency
- done b/f R.T
- less secure

10

else

Binary

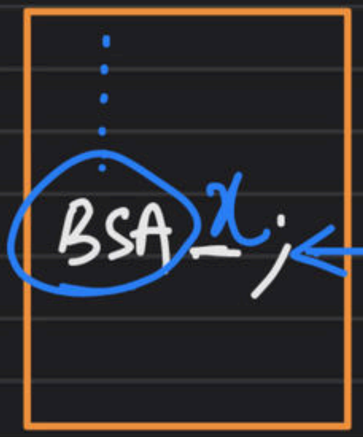
54KB



Dynamic

- Space efficiency
- Reusability of Libraries
- Time inefficiency
- Modifying Modules
- Linking carried out during R.T
- less secure

main (10KB)

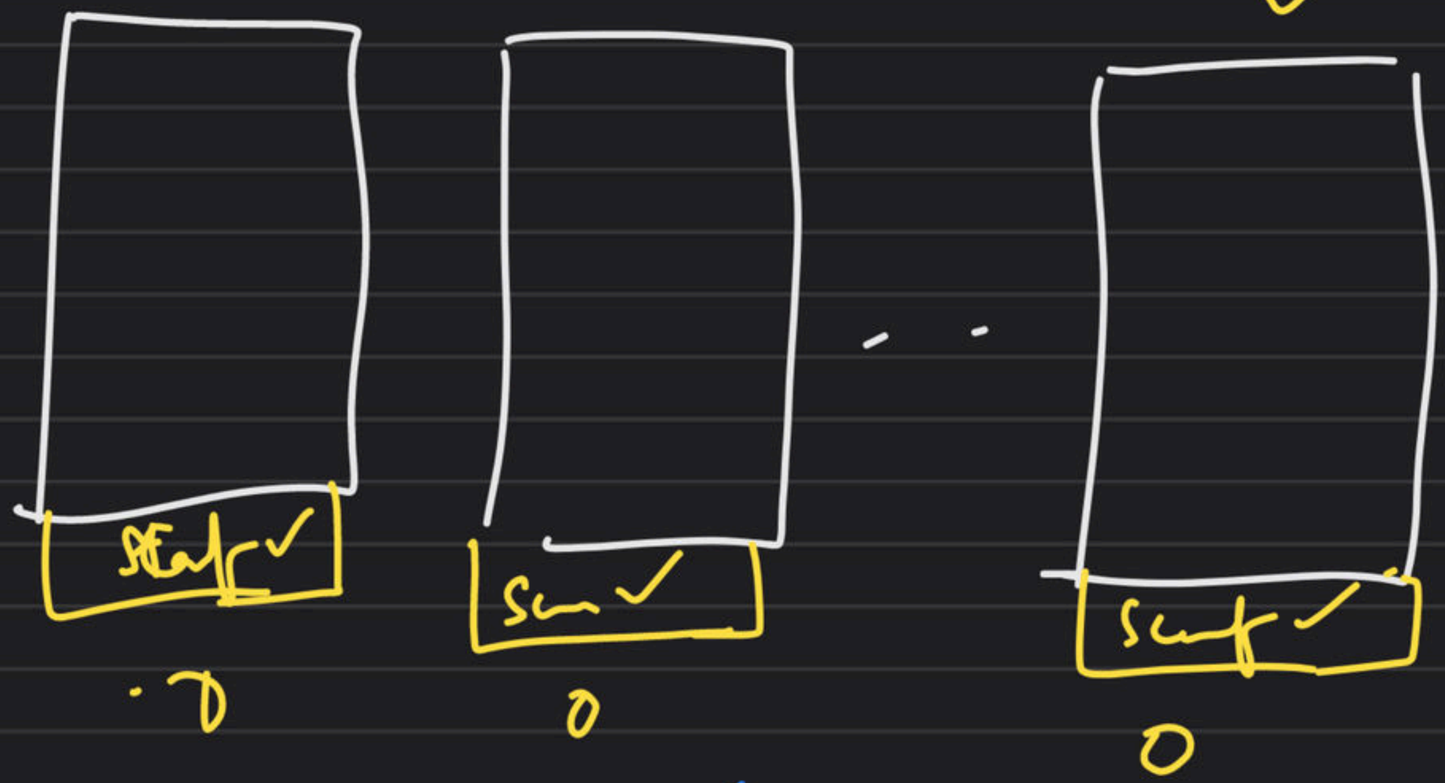


(... .DLL)

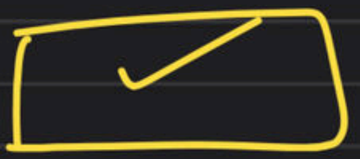
STUB <f, ...>

Linker 'f' x

Scanf ✓ ✓ Static linking ✓

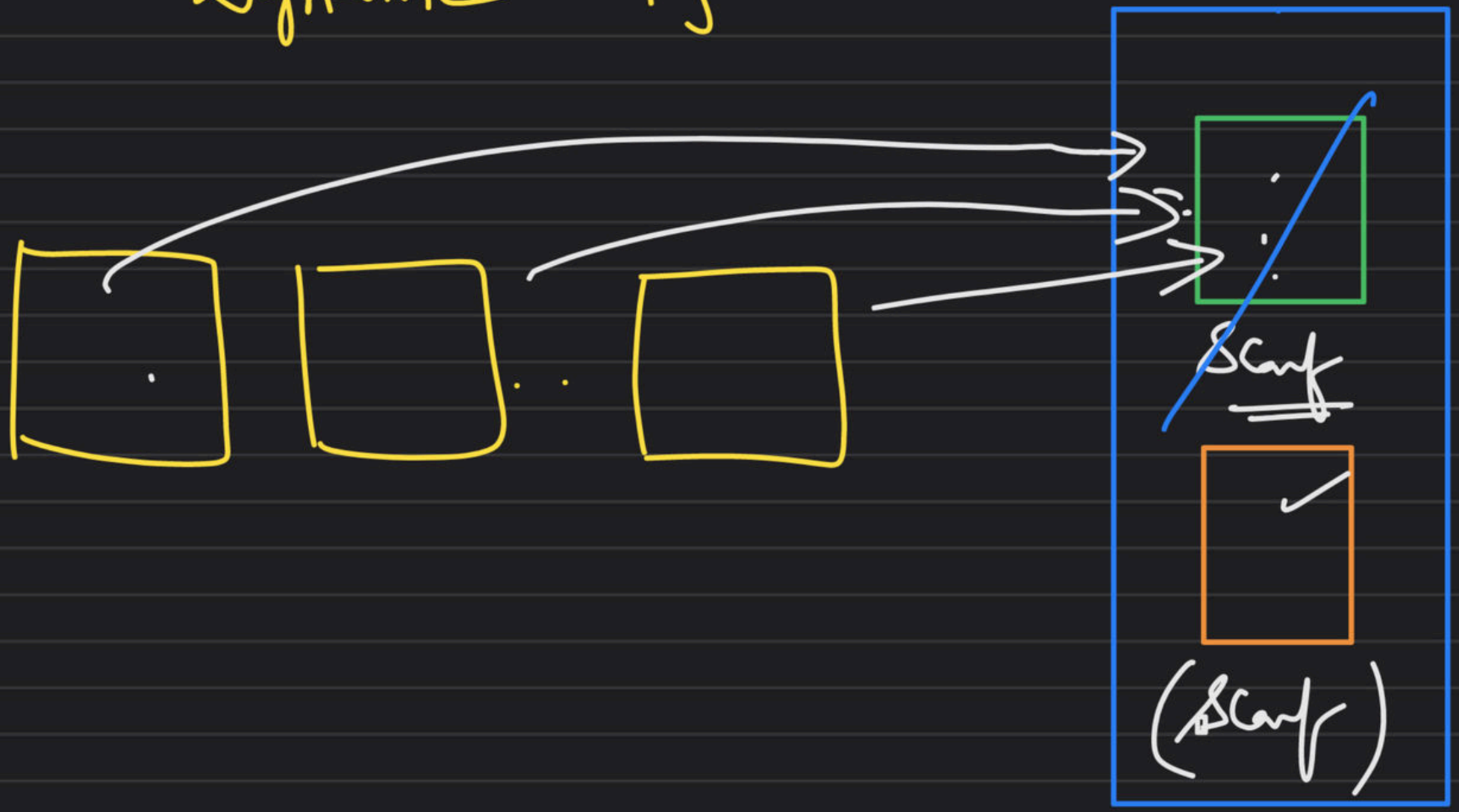


(Security)



Scanf(N)

Dynamic linking



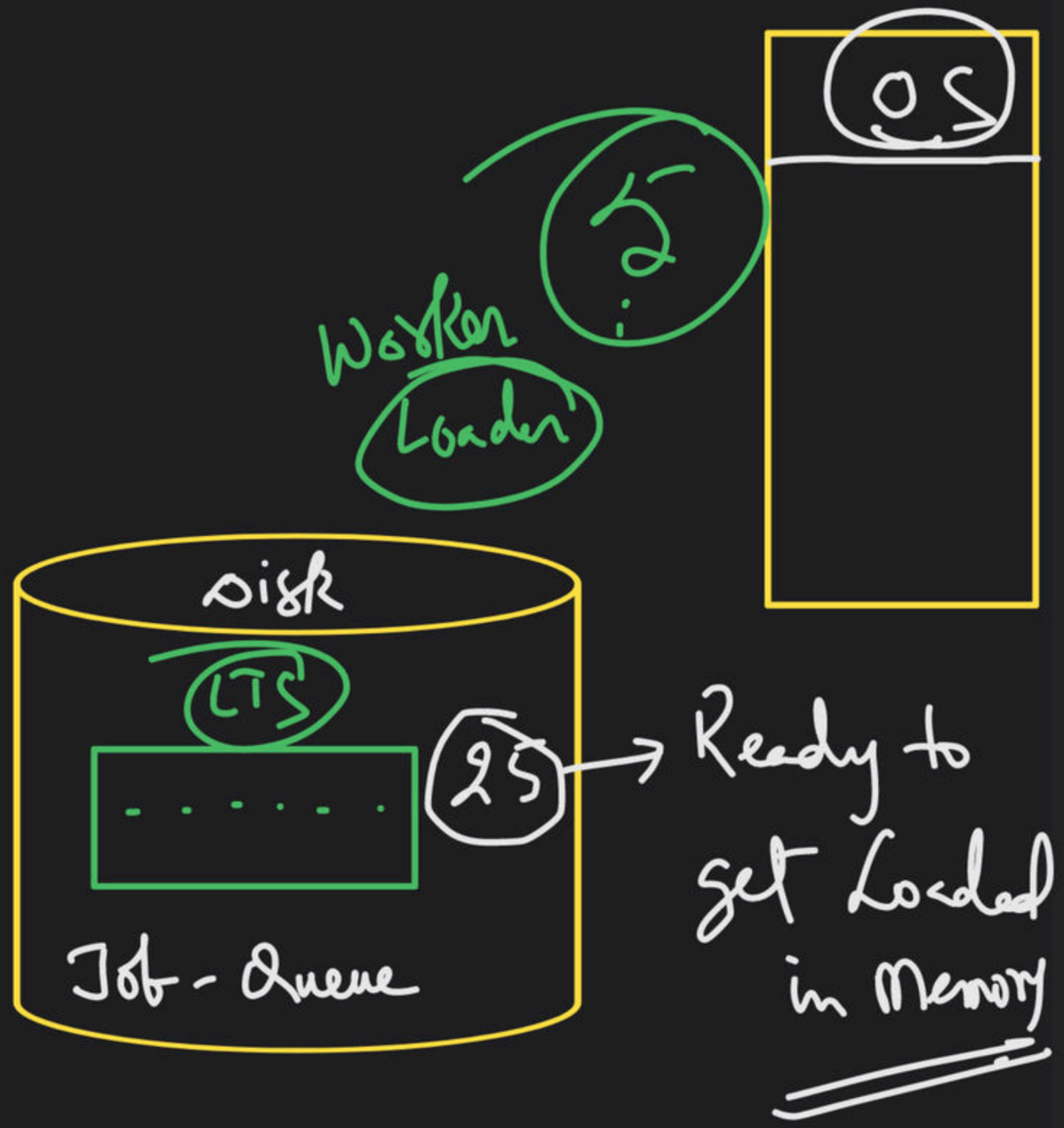
unacademy

▲ 1 • Asked by Arnob

sir you said that loader brings .exe from disk to main mem but doesn't LTS also does similar task when it brings new process from input queue (which is in disk) to ready queue (MM)

Scheduling

(Make a Decision)



unacademy

Address Binding :

int : 2B

: association of Program Instructions and data units to Memory Locations (Addresses) is known as Address Binding ;

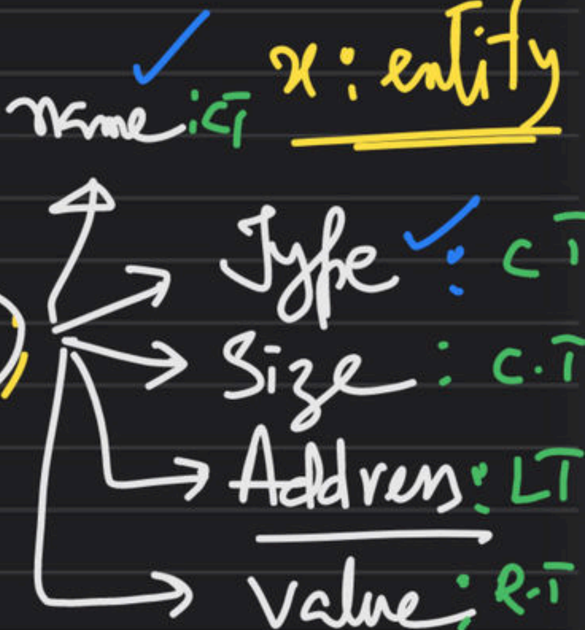
Binding (Association)

The time @ which Binding takes place is Binding time ;

(C.T R.T L.T)

```
main()
{
    int x;
    x = 5;
```

x : entity



```
main()
{
    int x, y, z;
    x = 1;      Ld  x, #1 : I1 : 2000
    y = 2;      Ld  y, #2 : I2 : 2002
    z = x + y;  Ld  r1, x : I3 : 2004
                Ld  r2, y : I4 :
                Add r1, r2 : I5 :
                Store z, r1 : I6 :
}
```

Code

Address

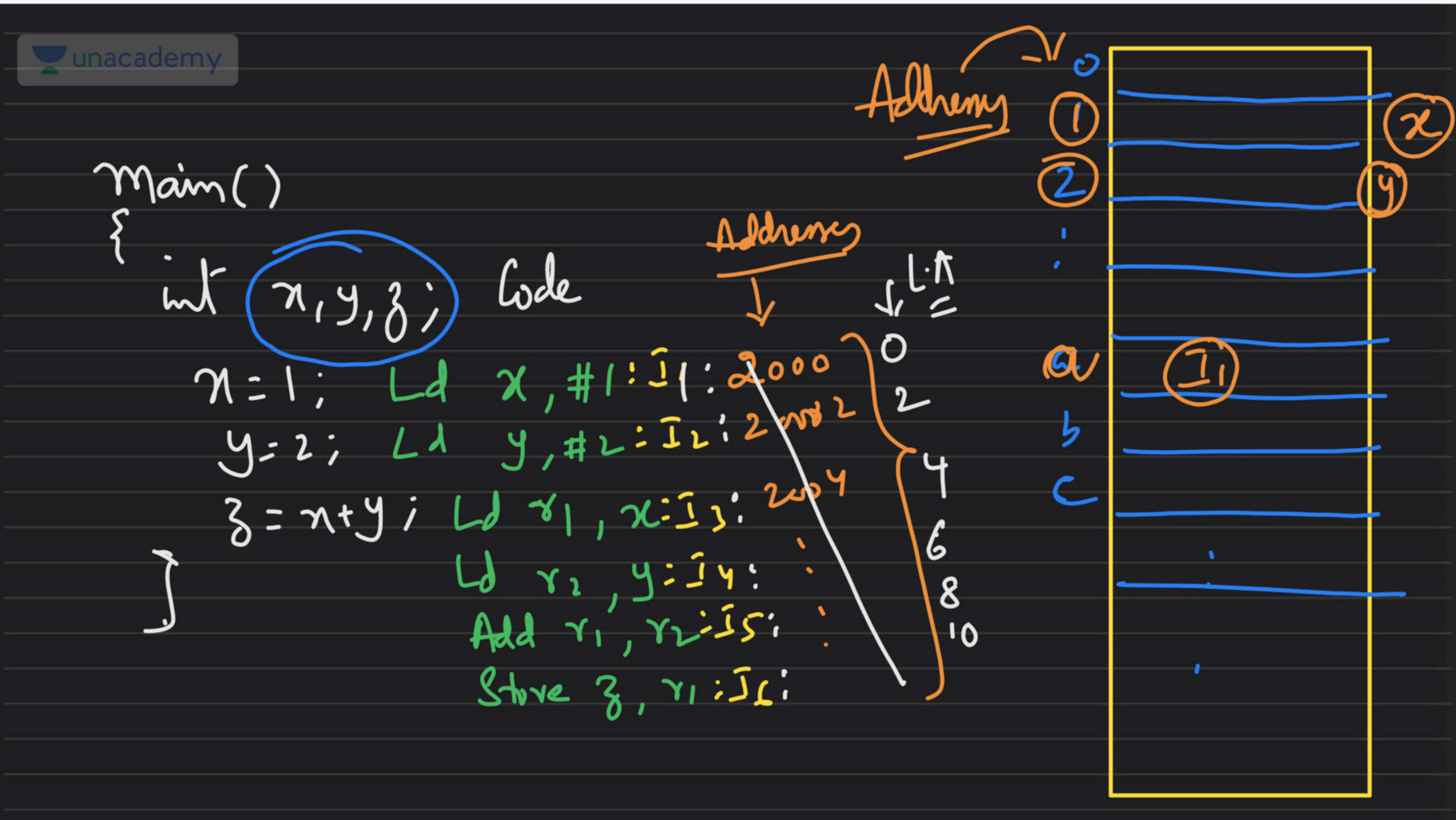
LC

0
2
4
6
8
10

Address

0
1
2
a
b
c

x
y
z
I1



Loader

Base Register

4000

OS

P.A's

I_1: 4000 + 0

I_2: 4000 + 2

I_3: 4000 + 6

⇒ Both C.T & L.T Addres Binding Schemes are "Static"

Types of Binding

(Static) Dynamic

During what time can Address Binding takes place:

1) Compile Time (CT): Compiler
2) Load Time (LT): Loader

*3) R.T Address Binding (Late Binding)

(Dynamic)

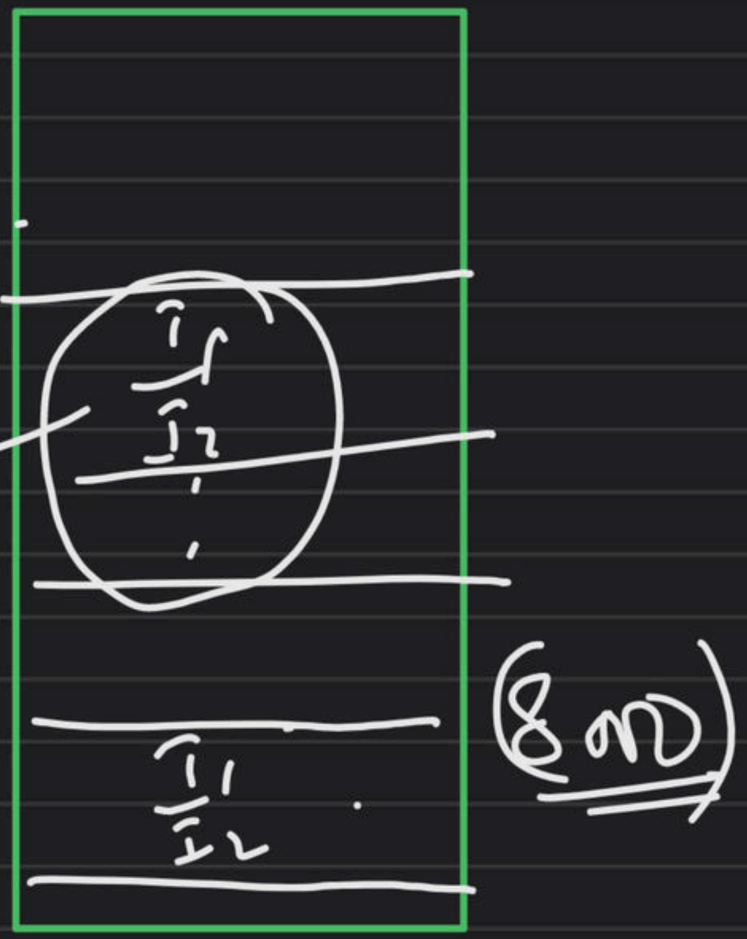
(Relocation)

Loader

6000

Base

6000

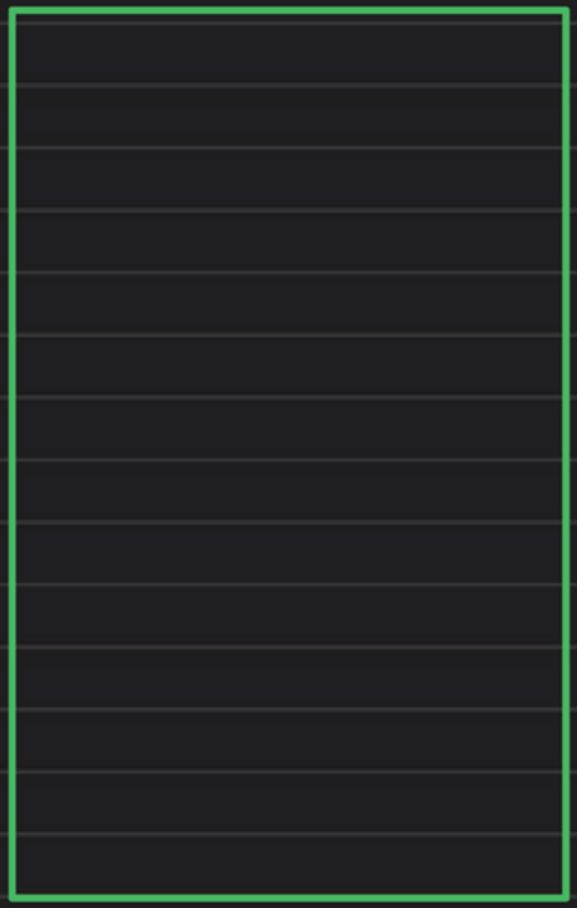


Swap out

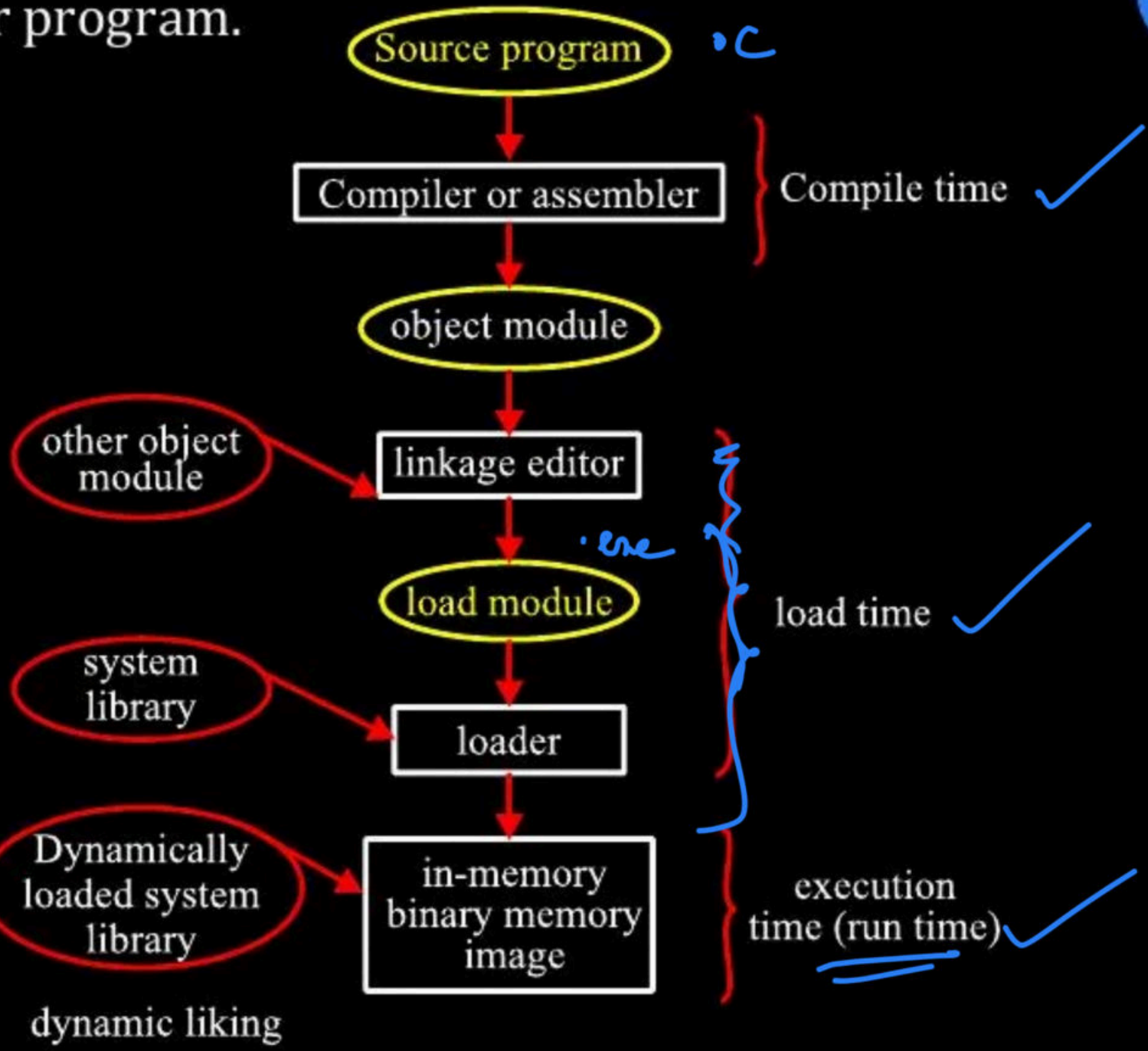
Swap in

Disk

(80K)



Multistep processing of a user program.



unacademy

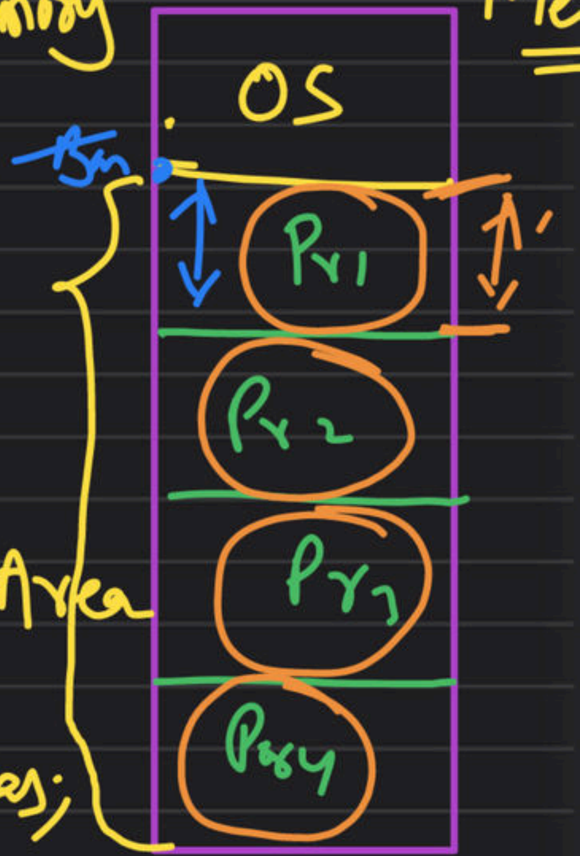
Memory Mgmnt. Techniques

Functions & Goals of Memory Manager

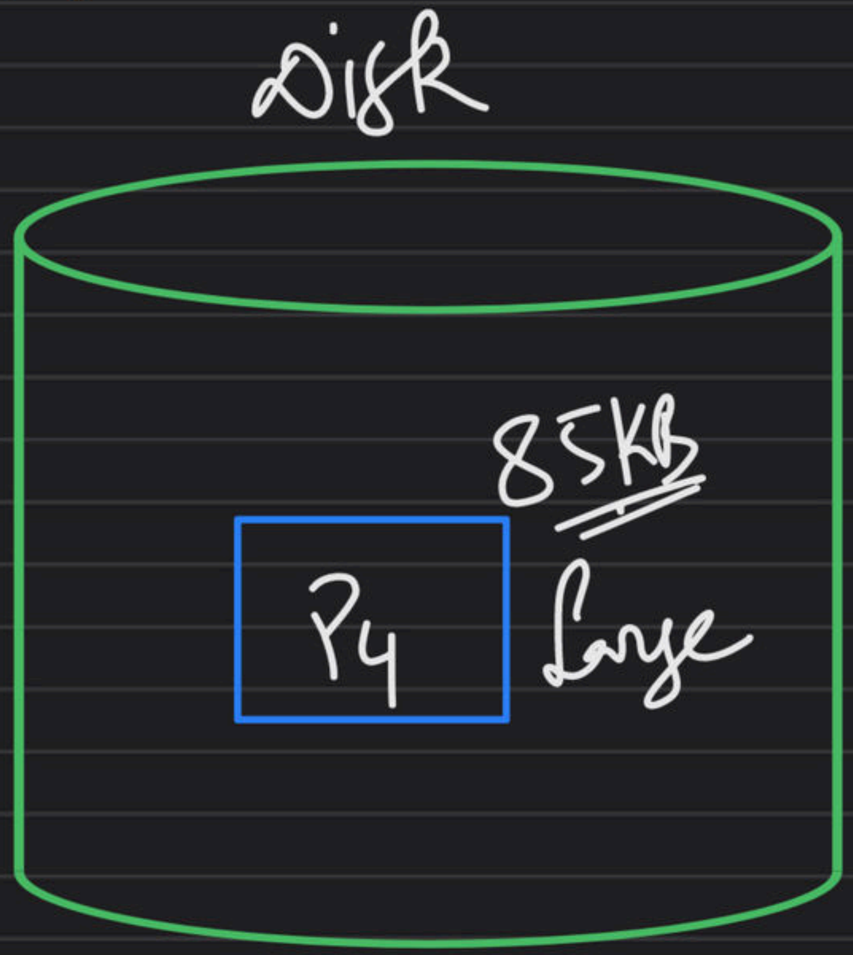
1) Allocation
2) Protection
3) Address Translation
4) Free Space Mgmnt;
5) Deallocation

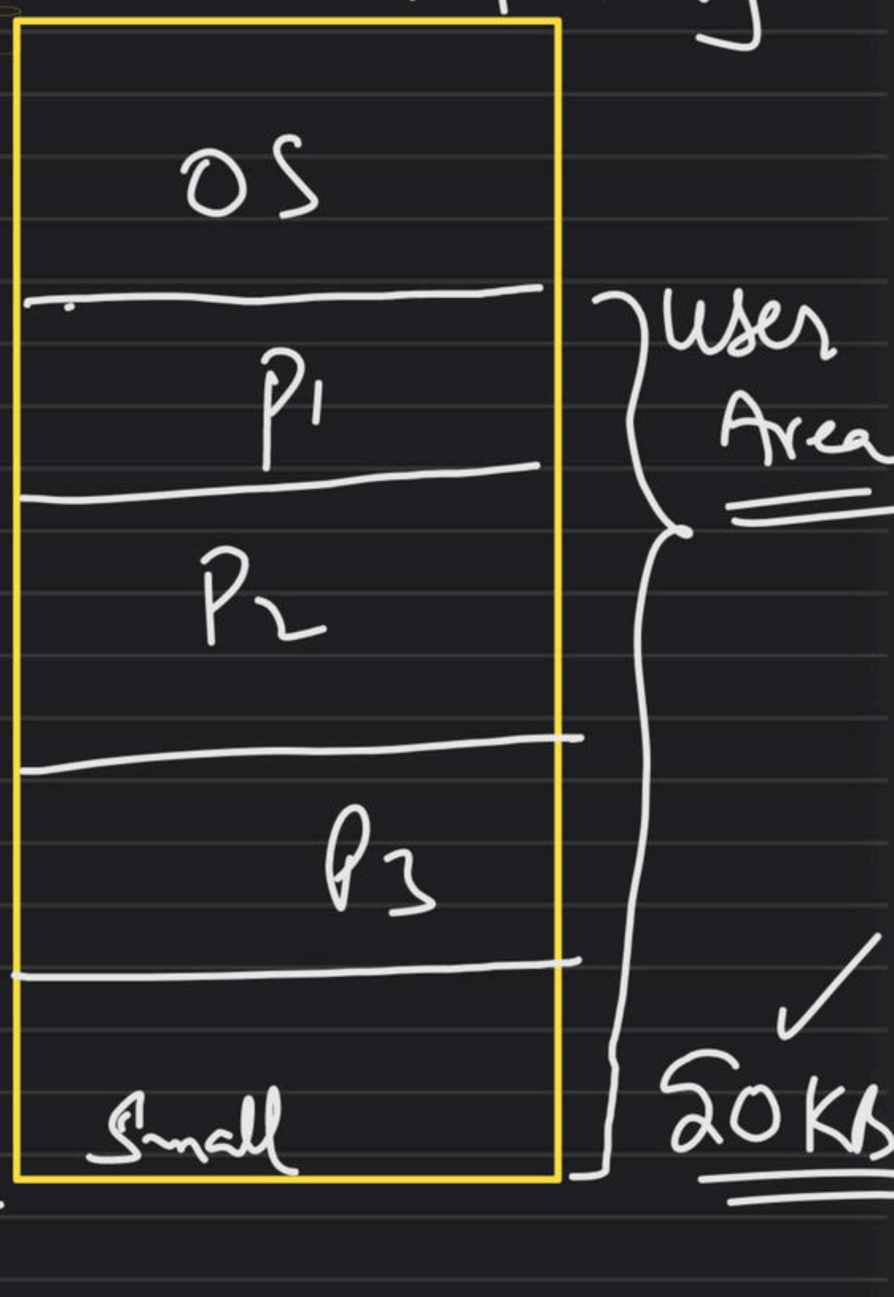
a) Effective utiliz. of Memory (less wastage) ⇕ (fragmentation)

b) Manage execution of larger Program in Small Memory Areas; (VM + overlays)

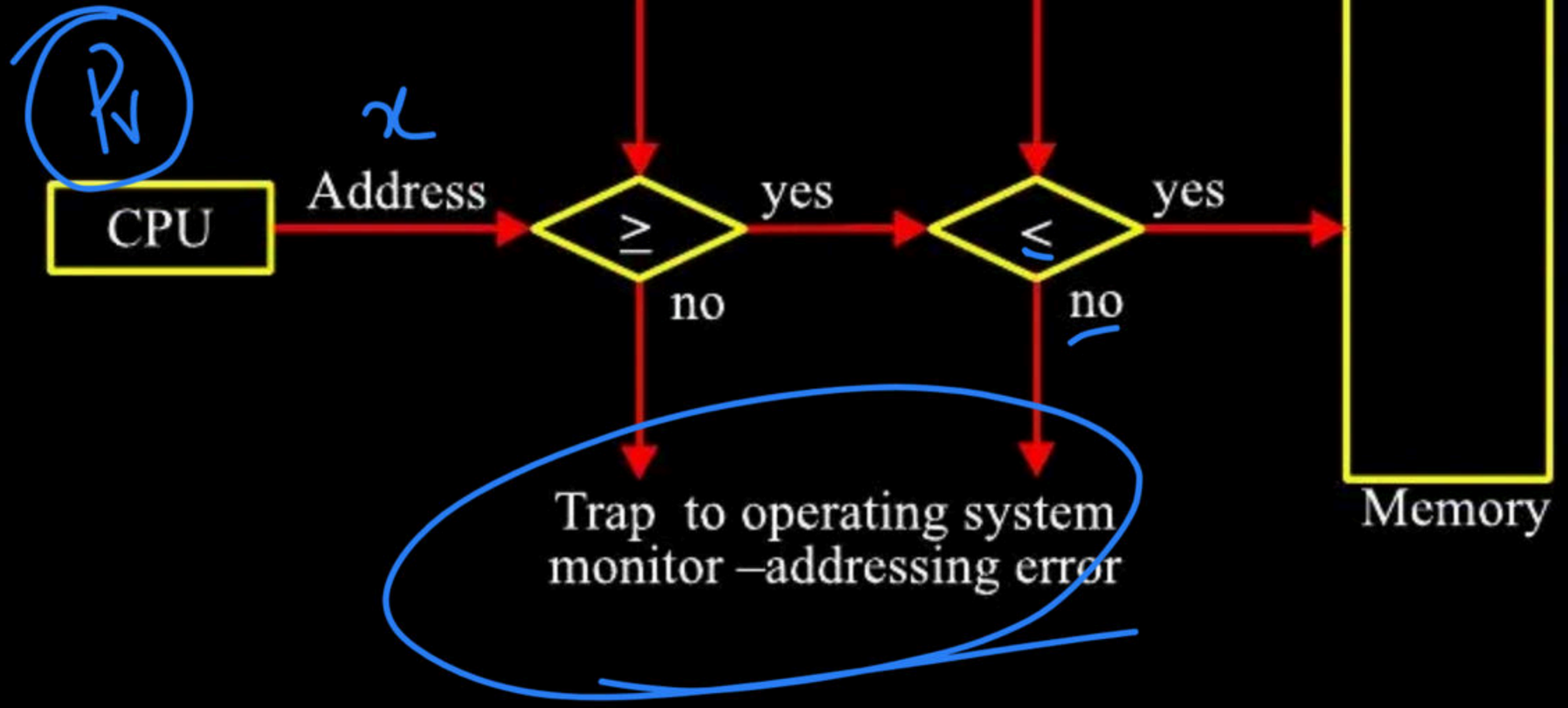


Adv:
run more programs
Incr. degree of M.P ↳ Throughput
?





Protection H/w :



P.A.S (MM)

2000
Base
Limit
500

2000
x
2500

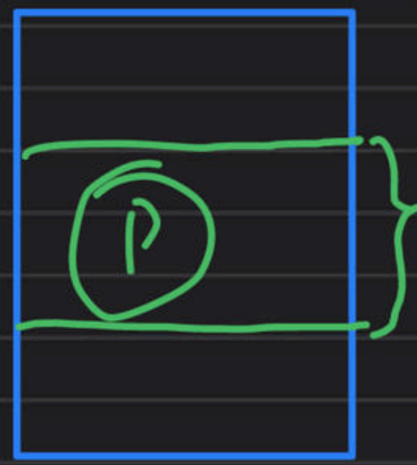
Memory Mgmnt. Techniques (classification)

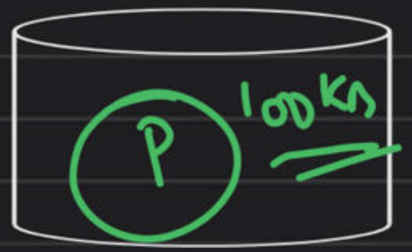
(i) overlays
(ii) Partitions — Fixed / variable
Buddy Sys.

Centralized (CG)

Contiguous

Memory

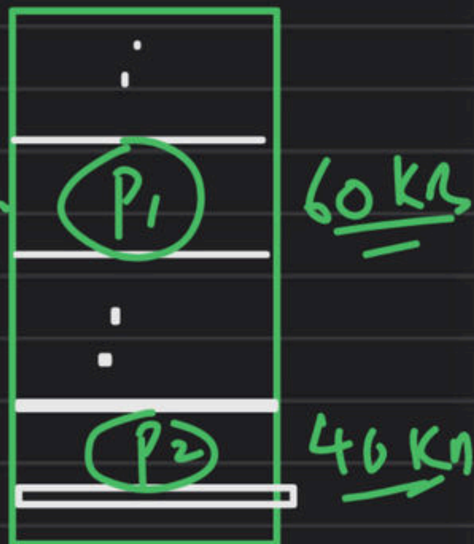




Distributed (NCG)

NON-Contiguous

(i) Paging
(ii) Segmentation
(iii) Seg-Paging
(iv) V-M



unacademy

1) Overlays : CG allocation :

Ex: 2 pass Assembler :

↳ Translator

↳ ALP → MLP

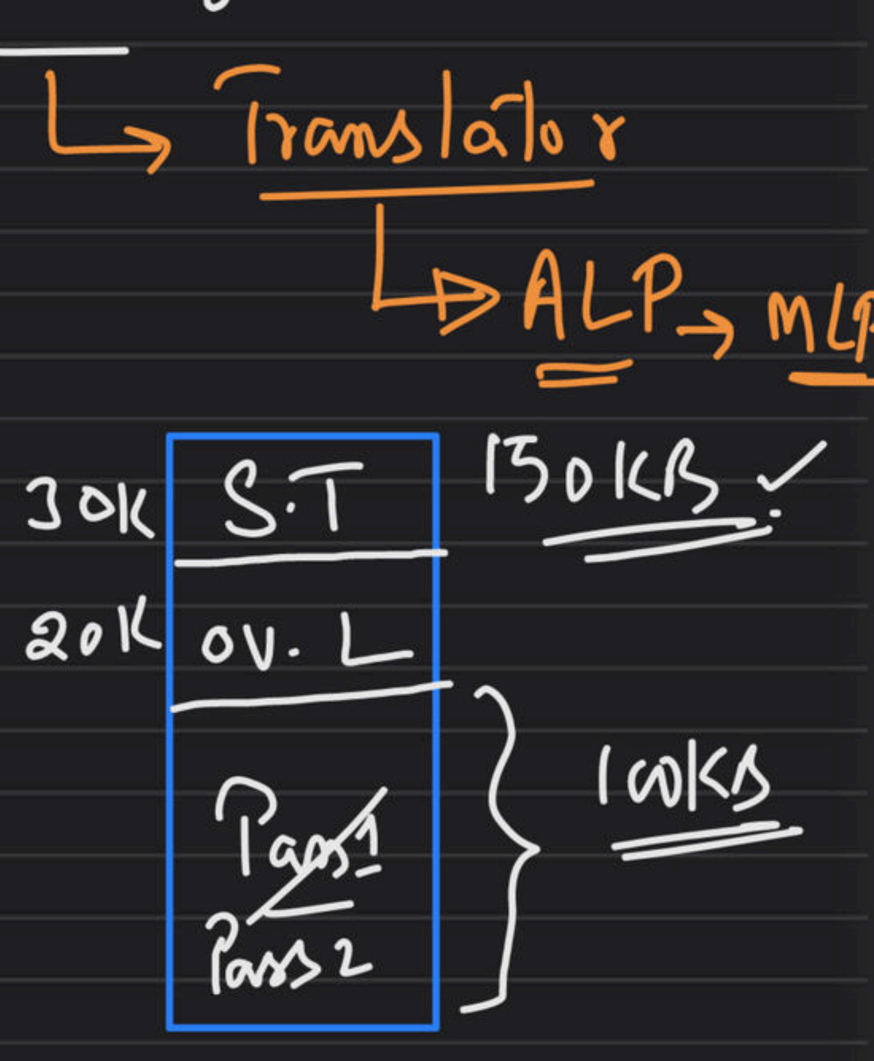
M/E

1) Pass 1 : 70 KB
2) Pass 2 : 80 KB

Symbol Table : 30 KB

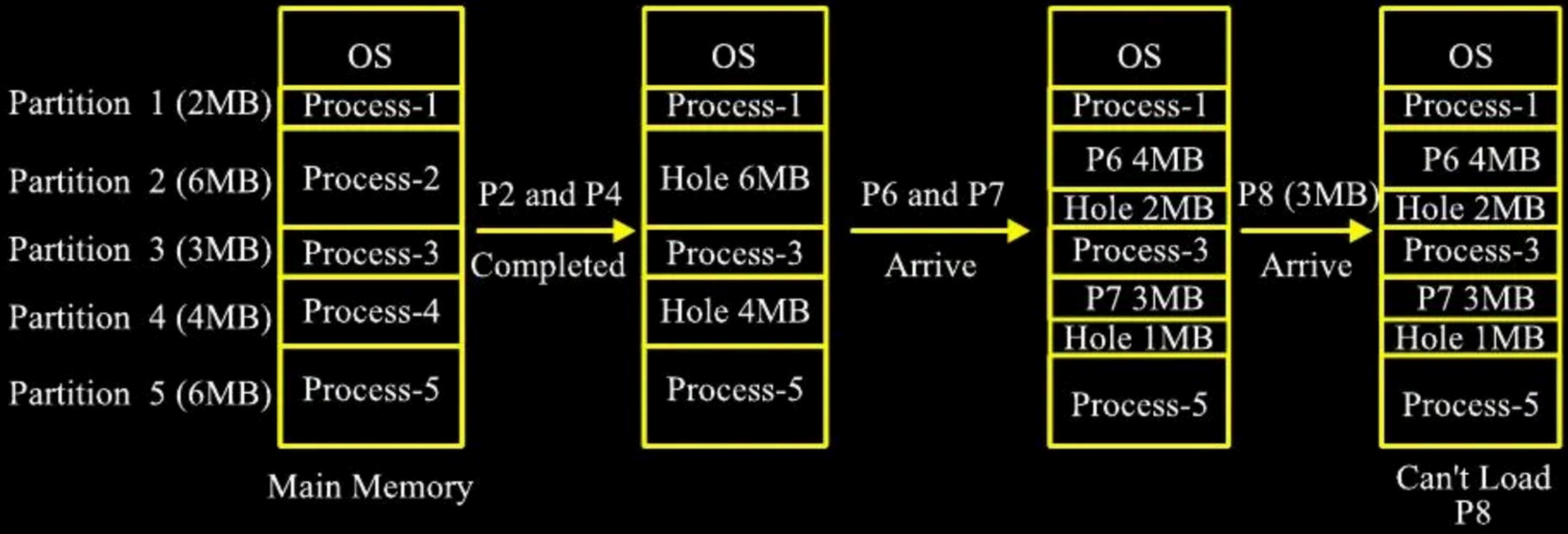
overlay Code : 20 KB

Total Size : 200 KB ✓



Limitation: is that overlaying is possible only when Program can be divided into Set of Independent Modules;

non-overlapping


Partition 1 (2MB)
Partition 2 (6MB)
Partition 3 (3MB)
Partition 4 (4MB)
Partition 5 (6MB)
OS
Process-1
Process-2
Process-3
Process-4
Process-5
Main Memory
P2 and P4
Completed
OS
Process-1
Hole 6MB
Process-3
Hole 4MB
Process-5
P6 and P7
Arrive
OS
Process-1
P6 4MB
Hole 2MB
Process-3
P7 3MB
Hole 1MB
Process-5
P8 (3MB)
Arrive
OS
Process-1
P6 4MB
Hole 2MB
Process-3
P7 3MB
Hole 1MB
Process-5
Can't Load
P8

Q. The capacity of a memory unit is defined by the number of words multiplied by the number of bits/word. How many separate address and data lines are needed for a memory of 4K x 16?

A 10 address, 16 data lines

B 11 address, 8 data lines

C 12 address, 16 data lines

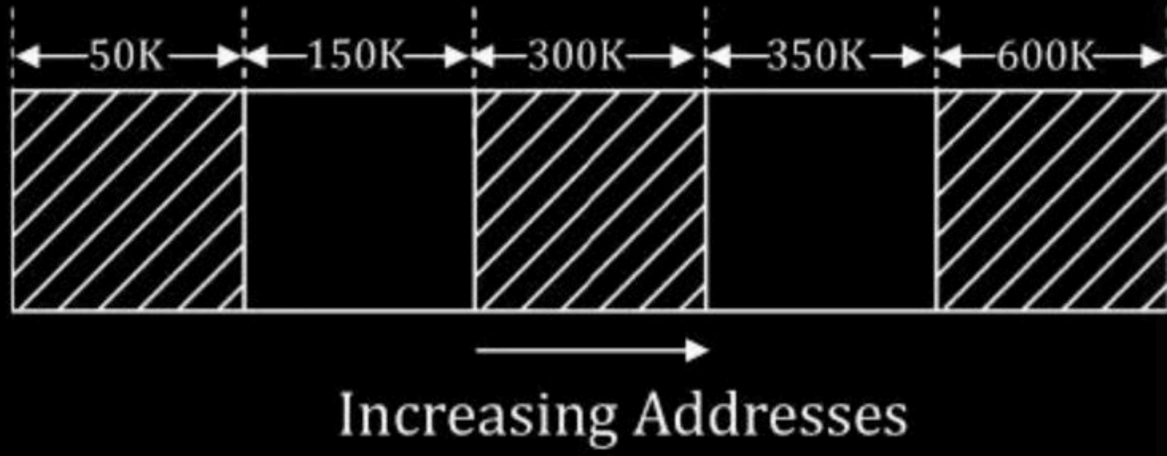
D 12 address, 12 data lines

Q.1

Consider a Memory System having 6 Partitions of sizes 200K; 400K; 600K; 500K; 300K; 250K. There are 4 Processes of sizes: 357K; 210K; 468K; 49K. Using Best Fit Allocation Policy, what Partitions are not allocated/ remains Unallocated?

Q. 2

Consider the following Memory Map in which blank regions are not in use and hatched regions are in use. Using Variable Partitions with no Compaction:



The sequence of requests for blocks of sizes 300K, 25K, 125K, 50K can be satisfied if we use:

A. Either first fit or best fit policy (any one)

B. First fit but not best fit policy

C. Best fit but not first fit policy

D. None of the above.

Q. 3

Consider a System with Memory of size 1000KBytes. It uses Variable Partitions with no Compaction. Presently there are 2 partitions of sizes 200K & 260K respectively.

(i) What is the allocation request of the Process which would always be denied?

A. 131 K

B. 151 K

C. 181 K

D. 541 K

(ii) The smallest Allocation Request which could be denied is:

A. 131 K

B. 151 K

C. 181 K

D. 541 K

Q. 4

Consider a System having Memory of size 2^{46} Bytes, uses Fixed Partitioning. It is divided into fixed size Partitions each of size 2^{24} Bytes. The OS maintains a Process Table with one entry per Process. Each entry has, two fields: First, is a pointer pointing to Partition in which the Process is loaded and Second, Field is Process ID(PID). The Size of PID is 4Bytes.

Calculate

(a) The Size of Pointer to the nearest Byte.

(b) Size of Process Table in Bytes if the System has 500 Processes.

Q. 5

Consider a System Using Variable Partition with no Compaction

Free holes	4K; 8K; 20K; 2K
Program size	2K; 14K; 3K; 6K; 10K; 20K; 2K
Time for Execution	4; 10; 2; 1; 4; 1; 8

Using Best Fit Allocation Policy and FCFS CPU Scheduling Technique, Find the Time of Loading & Time of Completion of each program. The Burst Times are in Seconds.

Q. 6

Consider allocation of memory to a new process. Assume that none of the existing holes in the memory will exactly fit the process's memory requirement. Hence, a new hole of smaller size will be created if allocation is made in any of the existing holes. Which one of the following statements is TRUE?

A. The hole created by Next Fit is never larger than the hole created by Best Fit

B. The hole created by Worst Fit is always larger than the hole created by First Fit

C. The hole created by First Fit is always larger than the hole created by Next Fit

D. The hole created by Best Fit is never larger than the hole created by First Fit

Q. In a variable size fixed partition-memory management scheme, internal fragmentation occurs when:

A. Sufficient memory is available to run a program, but it is scattered between existing partitions.

B. Insufficient memory is available to run a program.

C. The partition allocated to a program is larger than the memory required by the program.

D. A program is larger than the size of memory on the computer.

Q. Linker is given object modules for a set of programs that were compiled separately. What information need to be included in an object module?

MSQ

A. Object code

B. Relocation bits

C. Names and locations of all external symbols defined in the object module

D. Absolute addresses of internal symbols

(Physical Addresses)

Loader

Q. Dynamic linking can cause security concerns because

A Security is dynamic

B The path for searching dynamic linking is not known till runtime.

C Linking is insecure.

D Cryptographic procedures are not available for dynamic linking.

Q. An advantage of dynamic linking is that:

A. The segments that are not used in a run need not be linked into the process address space

B. It reduces execution time overhead

C. Debugging is simplified because programs are modular

D. The linker need not construct the known segment table

Q. A computer has 1 GB of RAM allocated in units of 64 KB. How many bits are needed if a bitmap is used to keep track of free memory?

A 2^{14}

B 2^{13}

C 2^{15}

D 2^{24}

Q. Let a memory have four free blocks of sizes 4K, 8K, 20K, 2K.These blocks are allocated following the best-fit trategy. The allocation requests are stored in a queue as shown below.

Request No	Request sizes	Usage Time
J1	2 K	4
J2	14 K	10
J3	3K	2
J4	6K	8
J5	6K	4
J6	10K	1
J7	7K	8
J8	20K	6

The time at which the request for J7 will be completed will be

A 16

B 19

C 20

D 37

unacademy

THANK YOU!

Here’s to a cracking journey ahead!