3.2 i) For $x = 1$

	Allocated	Maximum	Available	Need
P.A	10211	11213	00111	01002
P.B	20110	22210		02100
P.C	11010	21310		03000
P.D	11110	11221		00111

Initializing

Work = Available = 00111

Finish [x] = false where x = A, B, C, D

for Process D,

Need ≤ Work

⇒ Work = 00111 + 11110

= 11221

Finish [D] = True

Now, There exists no process with finish = false and Need ≤ Work

Final Finish Vector = F F F T

⇒ Not safe.

(i)

	Allocated	Maximum	Available	Need
P A	10211	11213	00511	01002
P B	20110	22210		02100
P C	11010	21310		10300
P D	11110	11221		00111

Initializing

Work = Available = 00511

Finish [x] = False for x = A, B, C, D

For Process D,

Need ≤ Work

⇒ Work = 00511 + 11110

= 11621

Finish [D] = True

For Process C,

Need ≤ Work

⇒ Work = 11621 + 11010

= 22631

Finish [C] = TRUE

For Process B, Need ≤ Work

⇒ Work = 22631 + 20110 = 42741

Finish [B] = TRUE

No process left with finish = false & Need ≤ Work. Final Finish Vector = TTTT

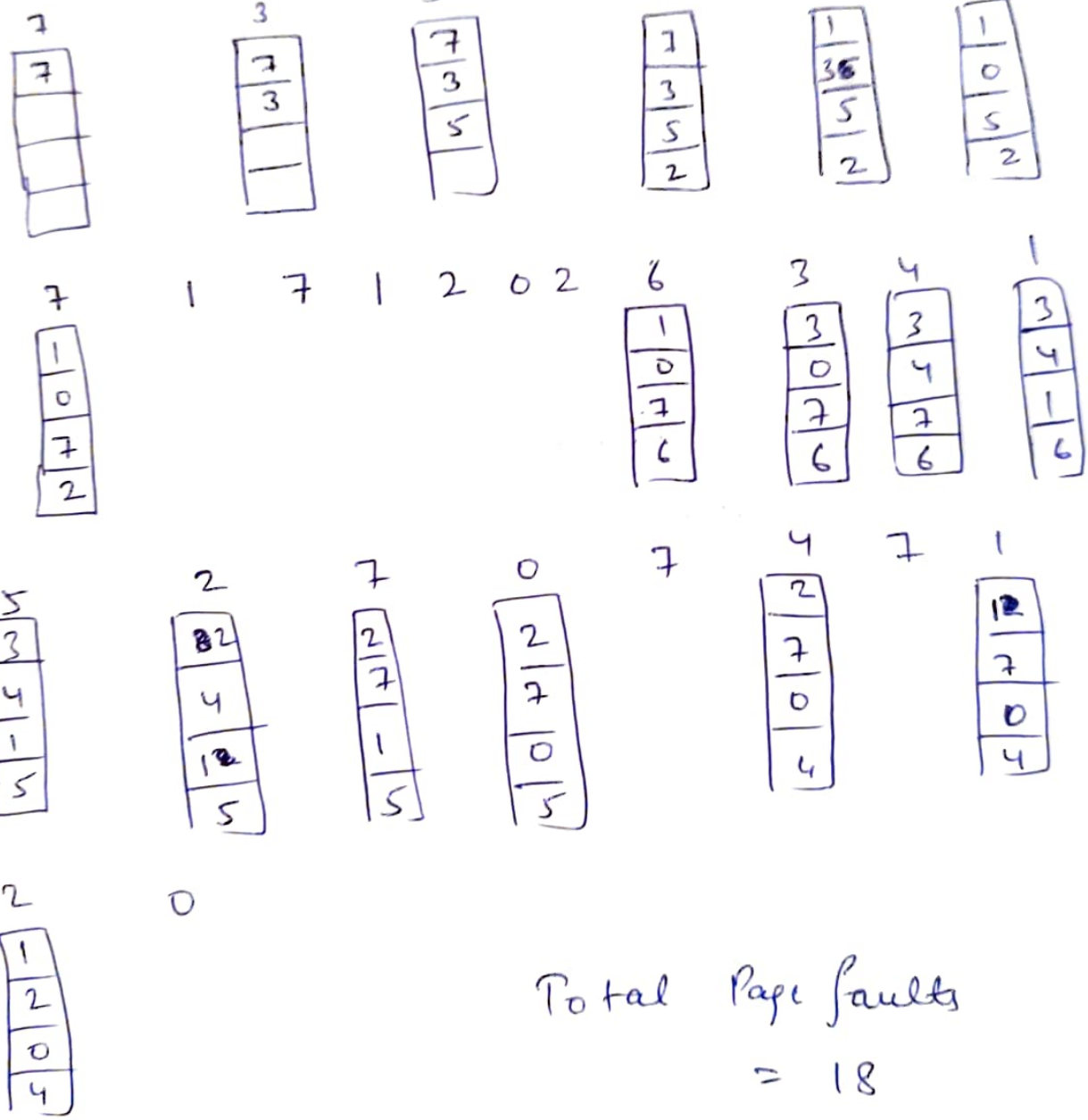
⇒ Not safe

4.2 Reference String given is

7 3 5 2 1 0 7 1 7 1 2 0 2 6 3 4

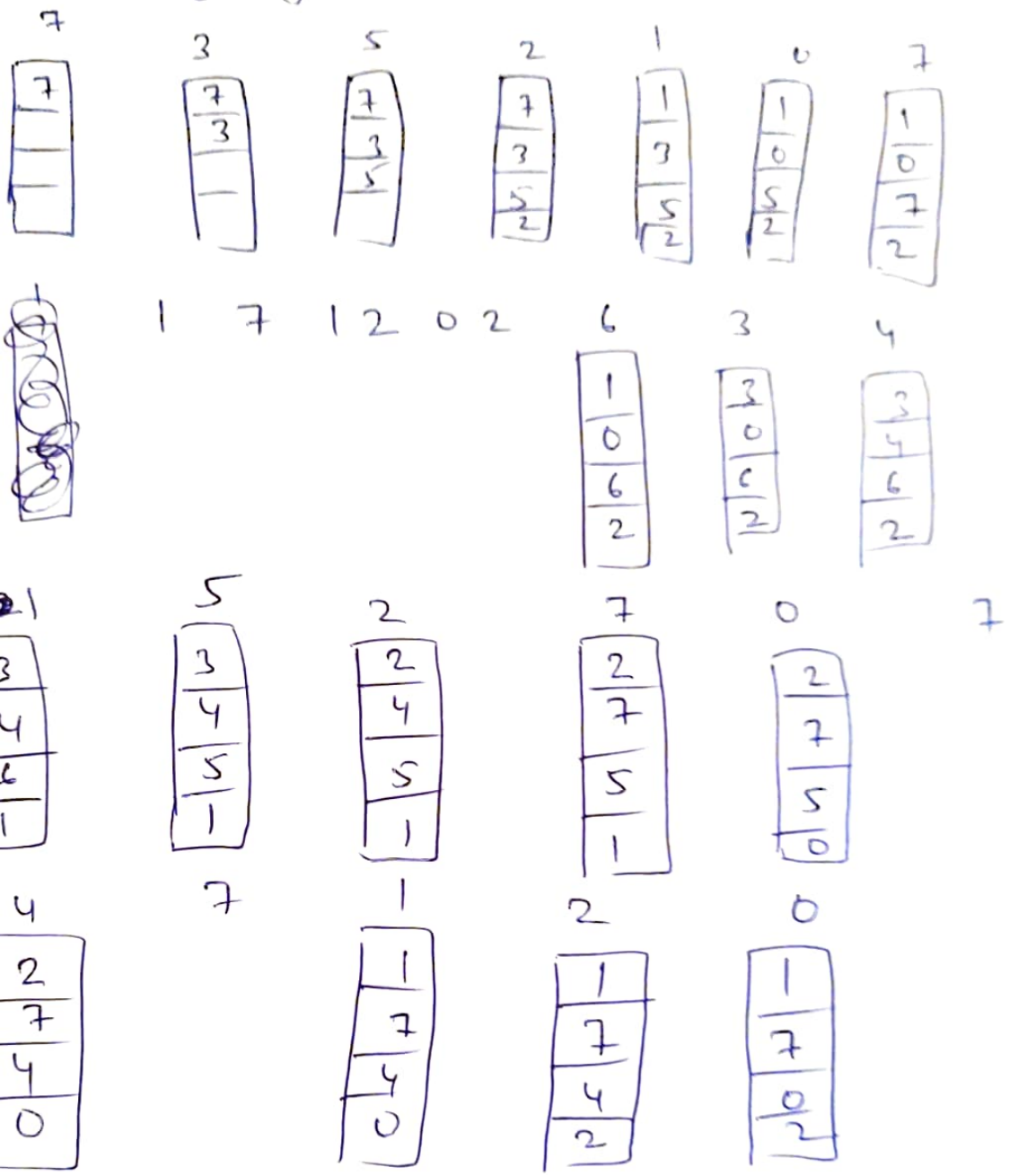
1 5 2 7 0 7 4 7 1 2 0

i) FIFO



Total Page faults = 18

ii) LRU Algo



Q:1

a) False, Validity of the entry is checked when it is loaded into TLB. Exception handler is called if found invalid.

b) TRUE, If the needed page is not in memory, Page fault occurs. OS fetches the page if it exists and process has the required permissions otherwise the process is aborted

c) ~~False~~,
TRUE, Same algo

d) ~~False, It is limited by the size variable used in fat, for each file~~

e) False, There exists atmax 1 inode of a file is system-wide open file table. Per-process file table of both processes will have different entries pointing to the same inode in system-wide open file table

d) TRUE, FAT has a table where each block can point to the next block of the file.
⇒ Only limit is the size of storage device.

f) TRUE, Interrupt handlers run in kernel threads

g) False :- Swapped out states contains ready processes which are swapped out of memory due to memory shortage

h) TRUE :- Soft links points to the filename which points inode of the file / folder, hence it can work across file systems

i) False :- Hard links dont work across different file systems. It knows nothing about where the data is stored in other file system.

j) FALSE :- FAT doesn't have support for hard links

19CS10050 Rupinder Goyal

3.1 Sum of Max. need $< m + n$.

Worst Case = All processes have max. resources and not finished.

⇒ Resources must be 1 less than the max need of the process.

→ Total available resources $< m + n - m = n$

Total resources $= n$

⇒ Deadlock cannot occur

4.1

i) Time = (L1 hit) 1×0.65 + (L1 miss) $0.35 \times$ [(L2 hit) 0.8×5 + (L2 miss) 0.2×100]

$= 0.65 + 0.35 \times (4 + 20)$

$= 9.05$ clocks

Rupinder Goyal 19CS10050

2) a) accessseats.P()

b) accessseats.V() & agentready.V()

c) agentready.P() & voterready.P()

d) accessseats.P()

e) accessseats.V() & voterready.V()

f. accessseats.V()