Name: Shyed Shahriar Housaini
ID: MCE 07905536; Subject: Operating System

Ans. to the q. no- 1(a)

1) Deadlock cannot occur in this case, because preemption exists.

2) In this case indifinite blocking may occur, a process may never get all the resources it needs.

(P.T.O)

Ans. to the q. no-1 (b)

Need table is
(Need = Max - Allocation)

A	B	C	D
3	3	3	2
2	1	3	0
0	1	2	0
2	2	2	2
3	4	5	4

T_0 & T_1 does not fulfill the need $\leq$ work (= Available).

T_2 is true for Need $\leq$ work

Ans. to the q. no - 2 (a)

Figure 1: allocation graph will not be in dead lock because

① T_2 will be completed and release resource

② T_4 will be completed and relece resource.

③ T_1 will get R_2 instance in addition to R_1 and will be completed.

④ R_1 will be released, T_3 will be completed by R_1 & R_2

Figure 2: (No. deadlock)

① T_3 and T_2 will get completed relese R_2 & R_1 instances. (T_3 & T_2 release Resources)

② T_1 will get R_1 & R_2 resources.

Ans. to the q. no 2(b)

Fragmentation	Contigious memory allocation	Paging
External	Holes in memory	Does not happens
Internal	Happens when required memory is less that allocated	Happens because of fragmentation
	Not used now	Used currently.

Ans. to the q. no - 2(c)

Six memory partitions are 100, 170, 40, 205, 30, 185 (All MB in order)

Hole sizes are 200, 15, 185, 75, 175, 80 (MBs in order)

First Fit allocation:

200 → 205
15 → 100
185 → 185
75 → 100
175 → can tot be allocated
80 → 170

Best Fit Allocation:

200 → 205
15 → 30
185 → 185
75 → 100
175 → ~~18~~ can not be allocated
80 → 170

Worst Fit allocation:

200 → 205

15 → 185

185 → can not be allocated

75 → ~~[illegible]~~ 170

175 → can not be allocated

80 → 100

Ans. to the q. no-3(a)

Logical address space of 2048 pages

page size = 4 KB

physical memory of 512 frames

logical address space size = 2^m

= no of pages x page size

= 2048 X 4096

2^m = 8,388,608

(Ans) m = 23 bits for logical address

Physical address size = 2^n

n is number of physical address

2^n = no of frames x frame size

= 512 X 4096

2^n = 2097152

(Ans) n = 21 bits for physical address.

Ans. to the q. no-3(b)

1. We have 2 memory access, 50 ns to access the page table and 50 ns to access the data in the memory

   total time = 100 ns.

2. Effective Access (reference) time

$$EAT = (.75 \times 2 + .25 \times 4)$$

$$= 2.5 \text{ ns}$$

~~Ans. to the q. no-3(c)~~

Ans. to the q.no-3(c)

Memory Protection during Paging:

① Memory protection is imposed by introducing a protection bit with all frame to show read only or read write is allowed.

② Valid-invalid bit added to each entry in the page table.

③ Use page table length register.

④ Any violation result in trap in kernel.

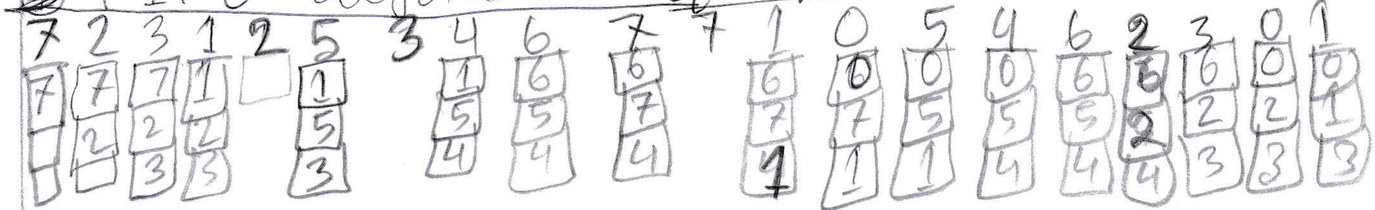
Ans. to the q. no-4(b)

Page reference string is

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

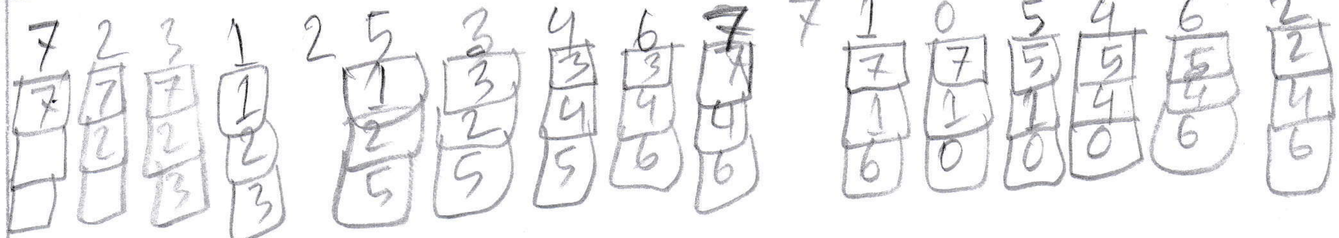
Demand paging with three frames,

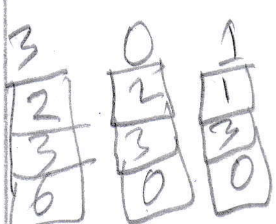
FIFO algorithm of Replacement:-



FIFO produces 17 page faults.

LRU Replacement:

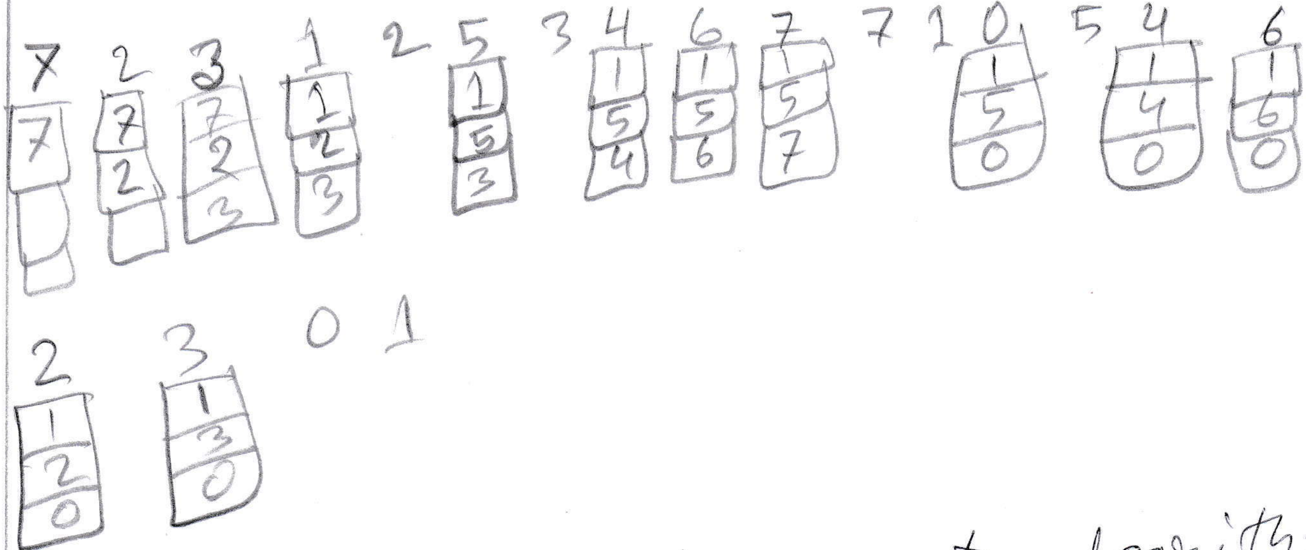




LRU produces 18 page faults.

LRU needs spacial hard ware and still slow.

Optimal page replacement algorithm:-



Optimal page replacement algorithm produces only 13 page faults. But this is very difficult to implement and used for benchmark.

In this case LRU performed not better than FIFO.

Ans to the q. no-4(a)

① When a page fault occurs the process will be blocked.

② Process will not change state

③ Thread will not change state.