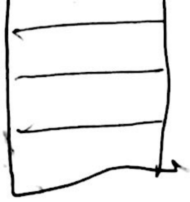
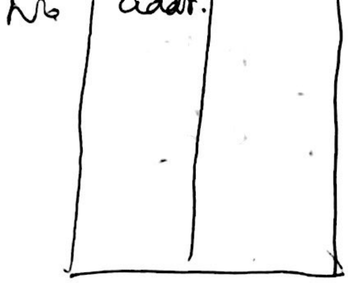




16/3/18

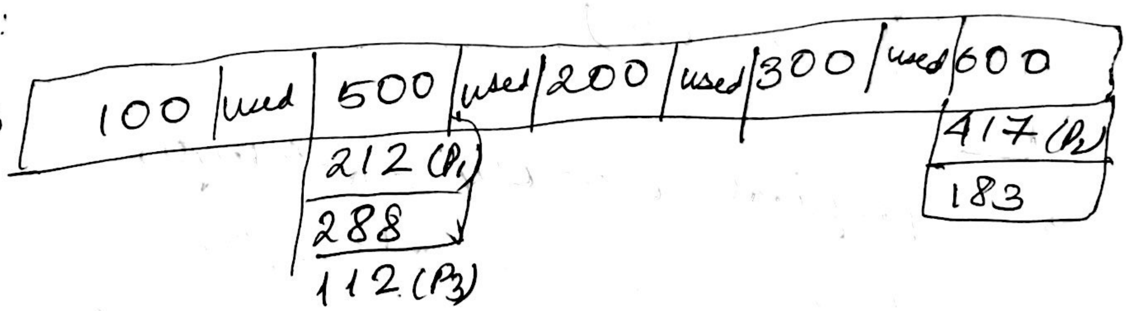
8\. 1) Given 5 memory partitions of 100 KB, 500 KB, 200 KB, 300 KB and 600 KB. How would the first fit, best fit and worst fit algorithms place processes of ~~200~~ 212 KB, 417 KB, 112 KB and 426 KB in order? Which algorithm makes the most efficient use of memory?

Ans:

First fit	Best fit	Worst fit
212 - 500	212 - 300	212 - 600
417 - ~~3~~600	417 - 500	417 - 500
112 - 200	112 - 200	112 - 300
426 -	426 - 600	426 -

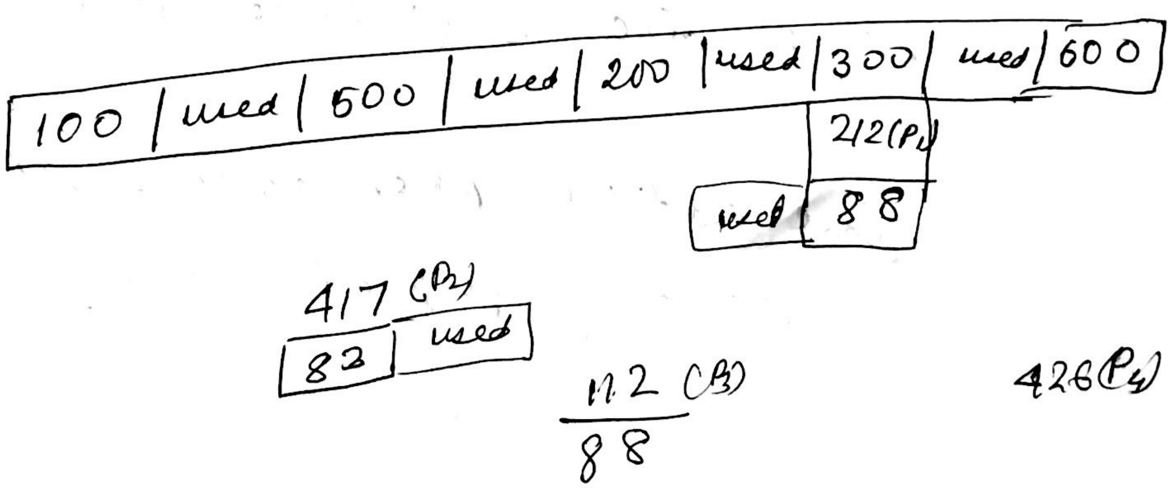
First fit:

Total: ~~3000~~



Rem:

Best fit:



worst fit:

100	used	500	used	200	used	300	used	600

$P_1(212)$

$\boxed{388}$

$P_2(417)$

$\boxed{83}$

$\frac{\cancel{P_3(112)}}{\cancel{188}}$

$\frac{P_3(112)}{188}$

Q2). On a ~~single~~ simple paging system with 2^{24} bytes of physical memory, 256 pages of logical address space and a page size of 2^{10} bytes, how many bits are in a logical address.

3) On a simple paging system with 2^{24} bytes of physical memory, 256 pages of logical address space & a page size of 2^{10} bytes, how many ~~entries~~ bits are needed to store an entry in the page table? (Assume each page table entry contains a valid/invalid bit in addition to frame no.)

Ans.

2). physical memory $= 2^{24}$ bytes

page size $= 2^{10}$ bytes

frame size $= 2^{10}$ bytes ($\because$ page size = frame size)

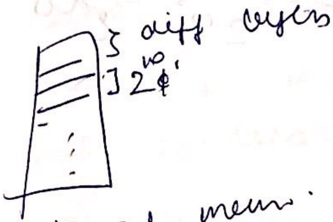
Logical address space contains $256 = 2^8$ pages.

No. of frames in physical memory $= \frac{2^{24}}{2^{10}} = 2^{14}$ frames

Page no. + offset gives the logical address

∴ 10 bits required to distinguish between different bytes

for $256 \rightarrow 2^8$ ∴ $8 + 10 = 18$



logical mem.

3) 14 bits req.d to identify 2^{14} frames.

frame size $= 2^{10}$ bytes

∴ bits to represent offset of frame $= 10$ bits.

∴ frame representation $= 24 - 10 = 14$ bits

∴ 14 bits ~~fram~~ for entry in the page table + 1 bit for valid/invalid bit

∴ <u>15 bits</u>

Q4) On a system that uses a 2-level page table has 2^{12} byte pages and 32-bit virtual addresses, the first 8 bits of the address serve as the index into the first level page table.

(a) How many bits specify the second level index?

(b) How many entries are in level 1 page table?

(c) How many entries are in level 2 page table?

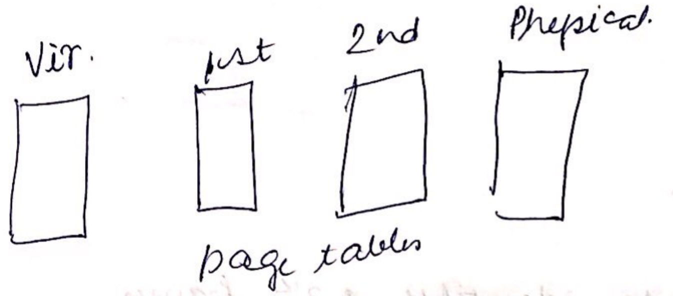
(d) How many pages are in virtual address space?

Ans : Page size = 2^{12} bytes

Logical / virtual address = 32 bits

No. of pages in the virtual address space

$= \frac{2^{32}}{2^{12}} = 2^{20}$



first 8 bits for first page table.

level 1 page table = 2^8.

$\therefore$ level 2 page table = $\frac{2^{20}}{2^8} = 2^{12}$

Bits required to specify 2nd level index = 12 bits

" " " " 1st " entries = 12 bits

5) In a simple paged system, associative registers hold the most active page entries and the full page table is stored in the main memory. If the references satisfied by the associative registers take 90 ns, and the reference through main memory page table takes 220 ns, what is the effective access time, if 60% of all memory references find their entries in the associative registers.

A : $(0.6 \times 90) + 0.4(90+220)$

=