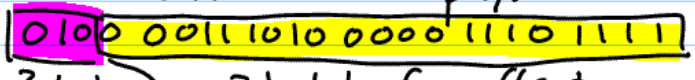

24-bit virtual addresses (16 MiB)
32 MiB physical memory
2 MiB pages
= 2^{21} bytes
process A
process B
physical memory
7
6
5
4
3
2
1
0
X
X
9
8
7
4
1
0
X
X
X
X
6
5
3
2
X
9
8
7
6
5
4
3
2
1
0

process A virtual address 43A0EF = physical addr 083A0EF



page 2 = frame 4

physical addr = 0100 00011101000001111011111

frame

B virtual address 43A0EF = physical addr

page 2 → frame 5

0A3A0EF

32-bit virtual addrs (4 GiB address space)
2 MiB pages (2^{21} bytes/page)

$2^{32}/2^{21} = 2^{11} = 2048$ entries in each page table

64-bit virtual addrs, 2 MiB pages → $2^{64}/2^{21} = 2^{43}$ entries per page table

need something to reduce size!

OOF!

multi-level paging : divide virtual addr into

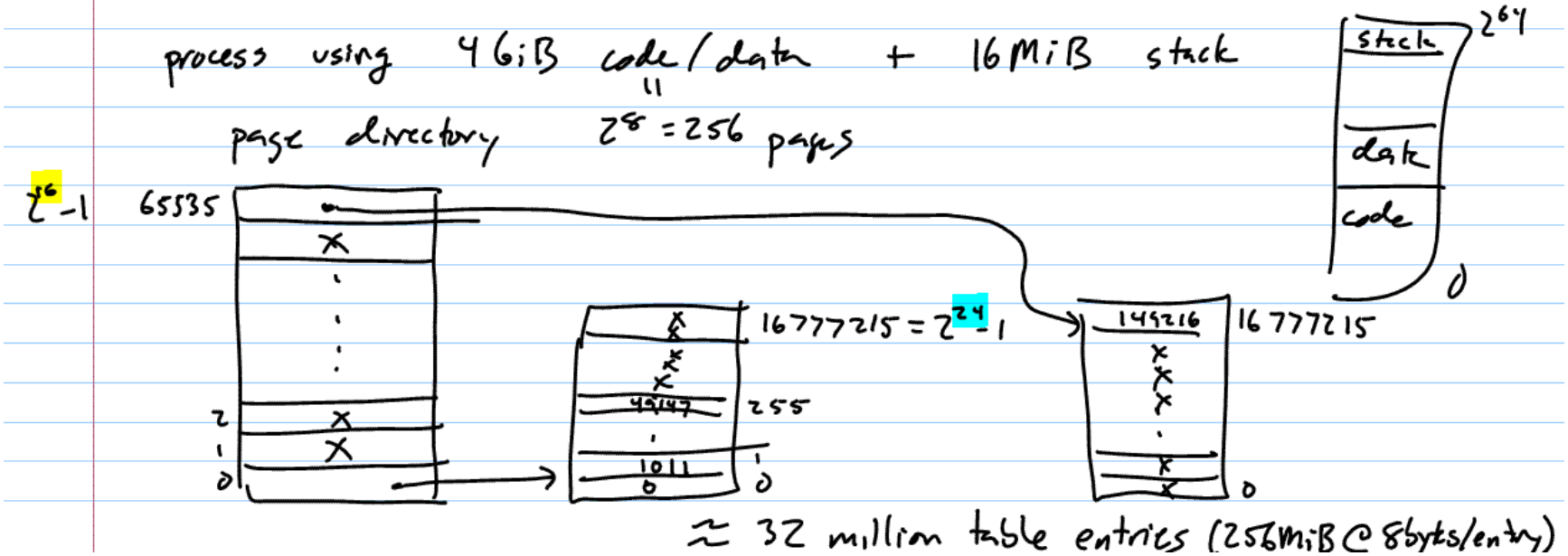
dir index	page #	offset

64-bit virtual addrs
16 MiB page size (2^{24} bytes/page)

dir	page #	offset
16	24	24

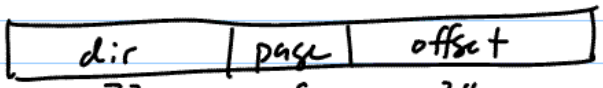
process using 4 GiB code/data + 16 MiB stack

page directory $2^{8} = 256$ pages



≃ 32 million table entries (256MiB @ 8bytes/entry)

alt split :



directory has ~4 billion entries

each 2nd level table has 256 entries