CS 326-02 Process Shared Memory

Project04 due Monday May 8th 11:59 pm

Kernel Free Pages

Page Directories

Process Shared Memory

Kernel Free Pages

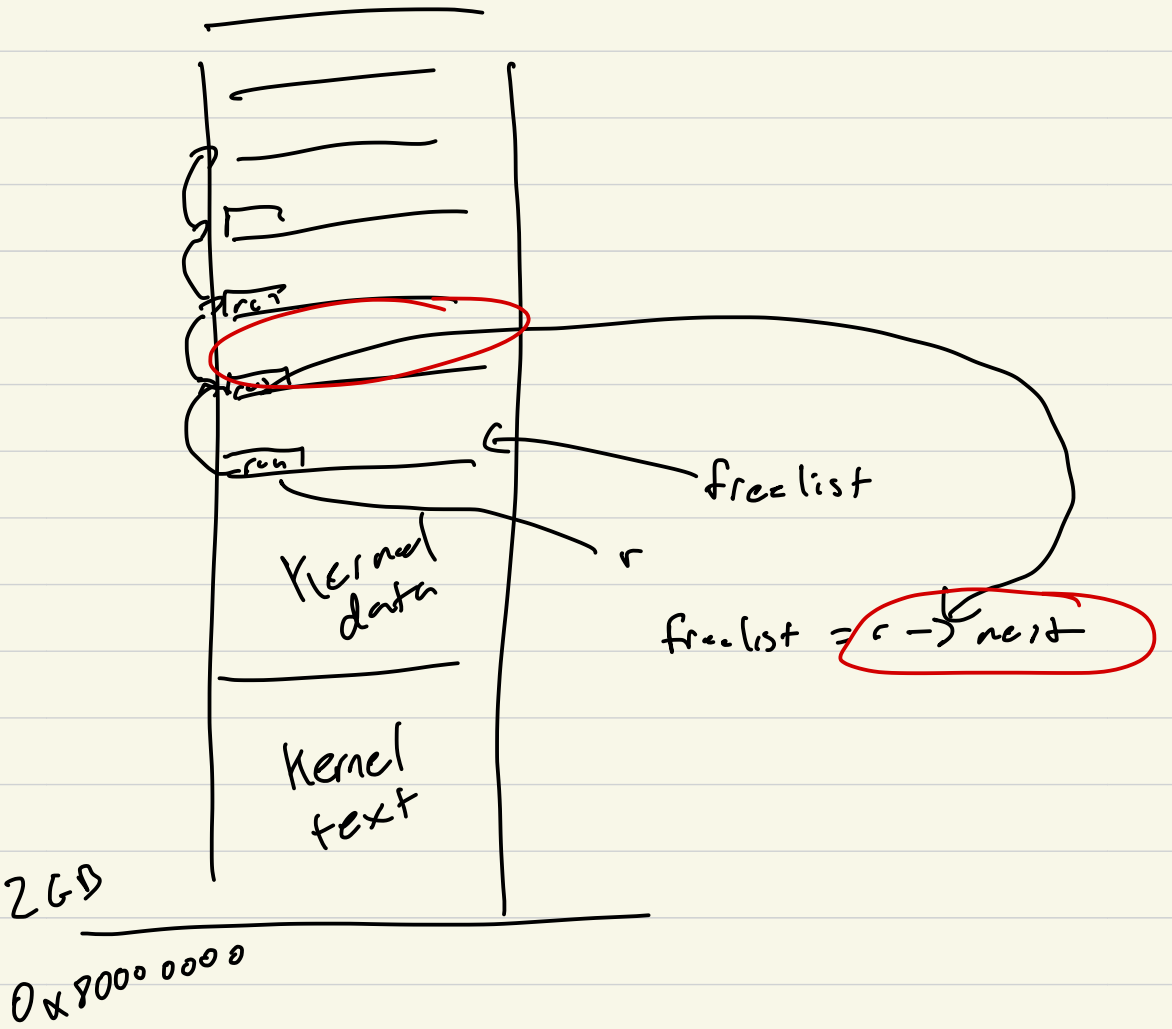
128 MB How many pages?

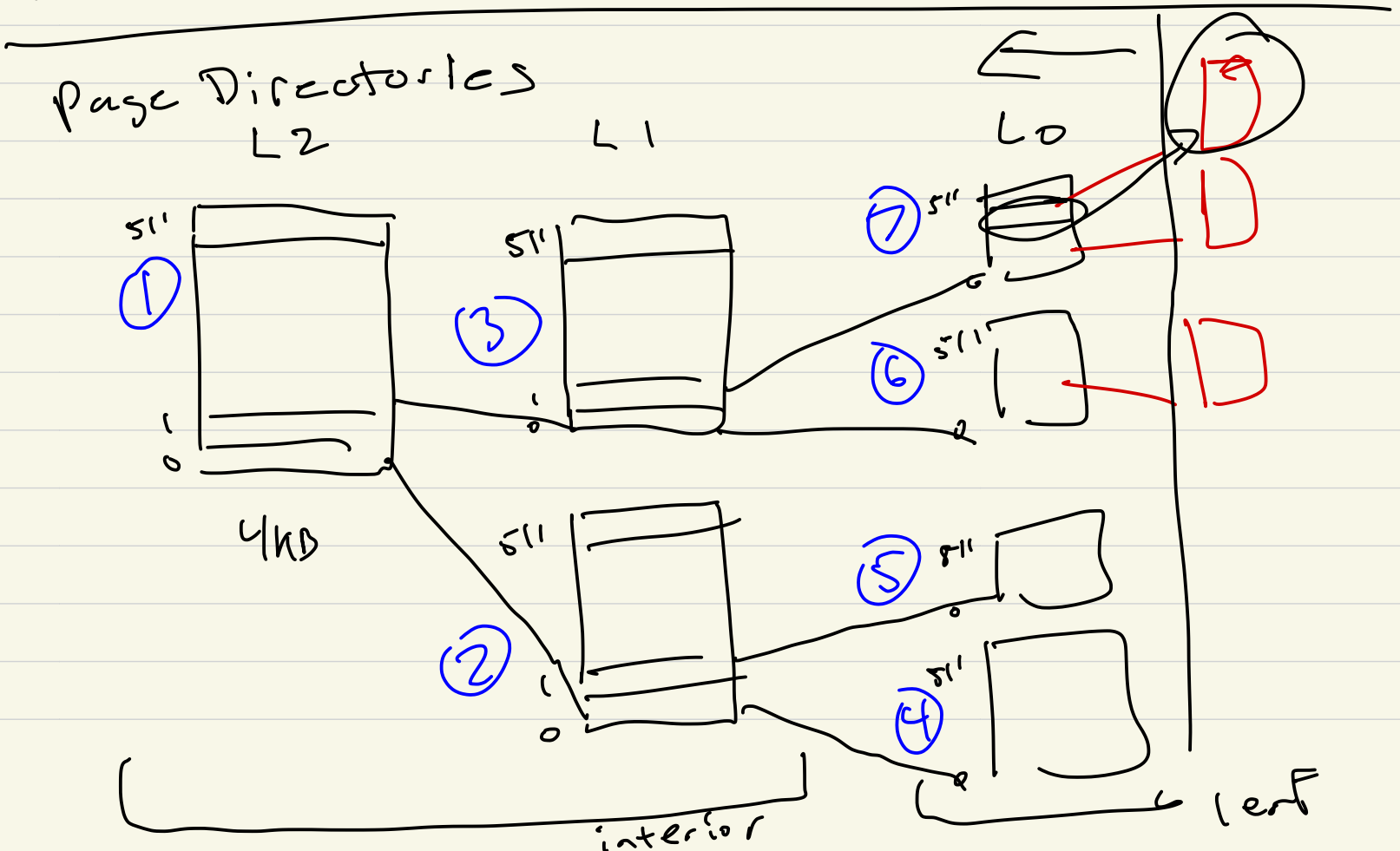
$1\ MB = 2^{20}$

$128 \times 2^{20} = 2^7 \times 2^{20} = 2^{27}$

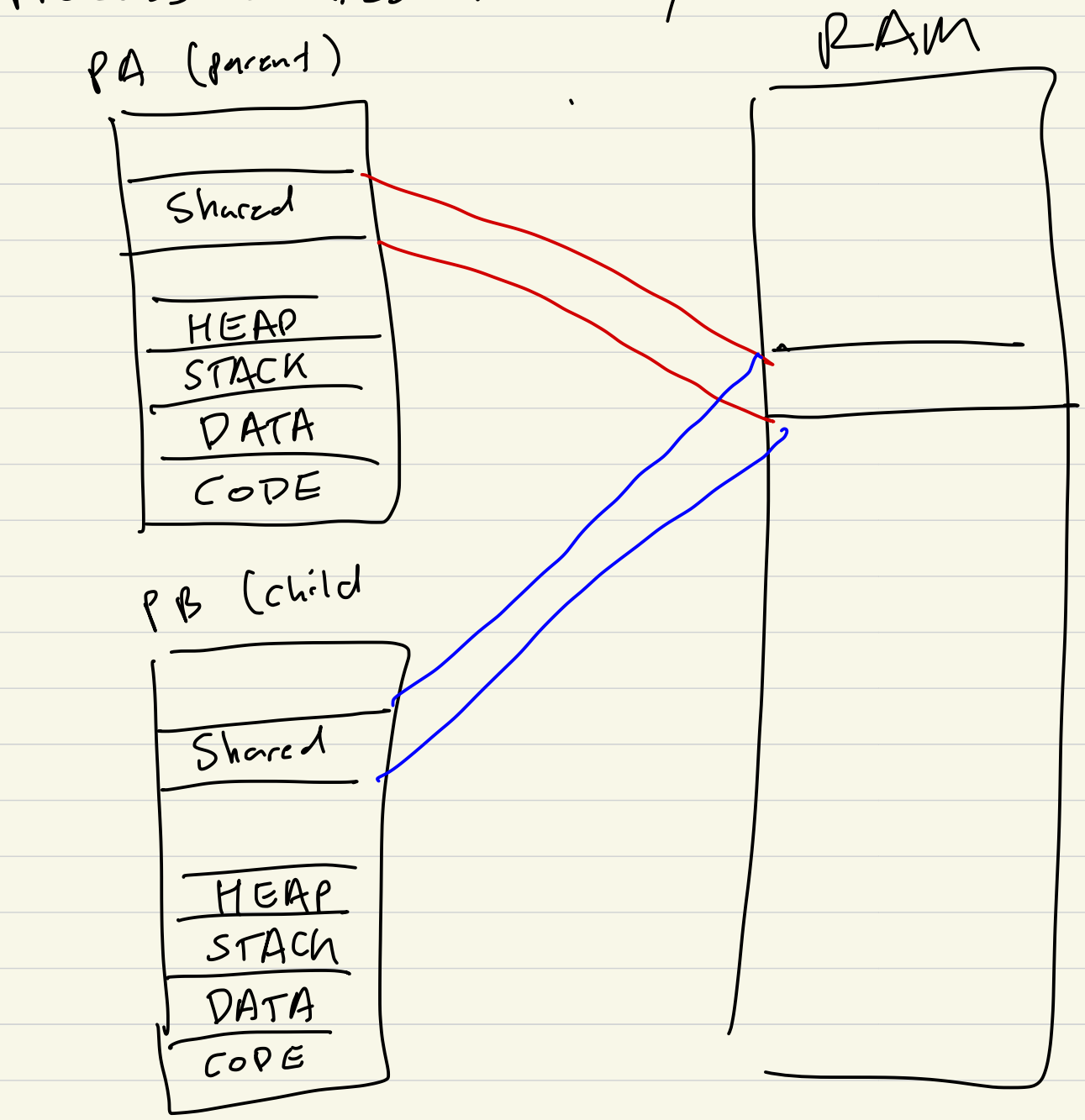
$2^{27} / 2^{12} = 2^{15} = 32768$

$32768 - 32553 = 215$ pages


freelist
r
Kernel data
freelist = r->next
Kernel text
2GB
0x8000 0000


Page Directories
L2
L1
L0
512
1
3
7
6
5
4
2
4KB
interior
leaf

Process Shared Memory



Things to Consider

smem(addr, size)

1. check that addr and size are multiples of 4096
2. Allocate phys pages with kalloc()
3. check conflicts in VA(addr)
4. mappages to addr in the process pagetable

Concept: a shared memory region will have a owner

In proc struct:

addr (UVA)
size
owner_pid

fork()

if parent has a shared memory region then have the child inherit the region.

map same virtual address region in childs pagetable

freeproc()

if proc has shared mem

if proc is owner

unmap and deallocate pages

else

unmap