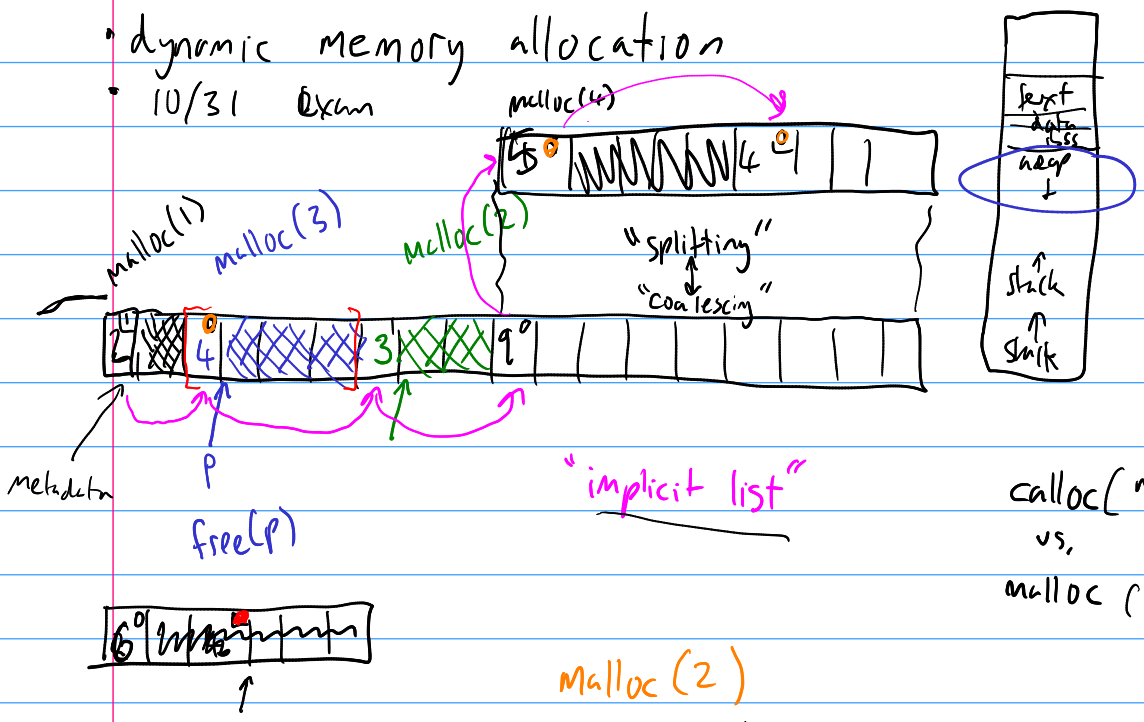
CS 3214 malloc: free lists

No class on Thursday (after 3:15)

- dynamic memory allocation
- 10/31 Exam



use-after-free
double free

calloc(n * s)
vs, "
malloc(v)

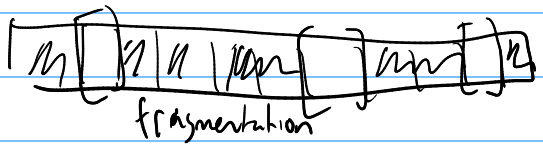
malloc(2)

allocation placement policy:

- → first fit:
- → best fit
- next fit

> speed of allocation / tput vs. > use of memory

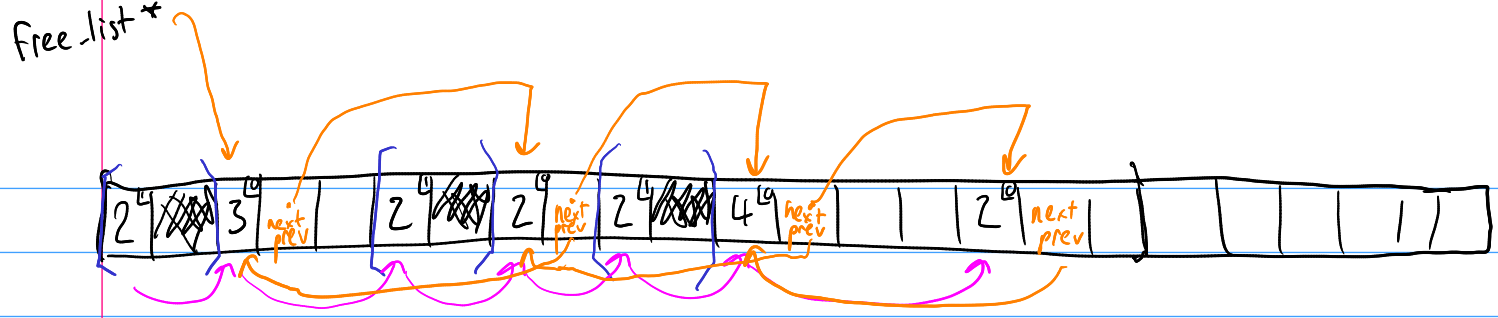
fragmentation

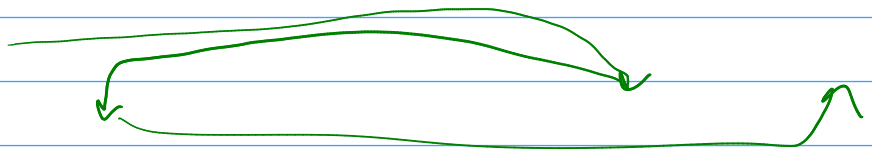


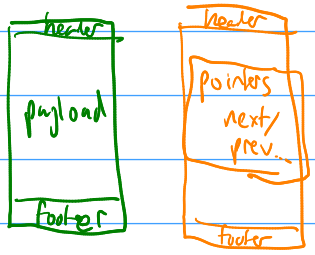
Simple malloc

+ simple
+ freeing is fast
- allocation can be linear in size of mem SLOW
all blocks

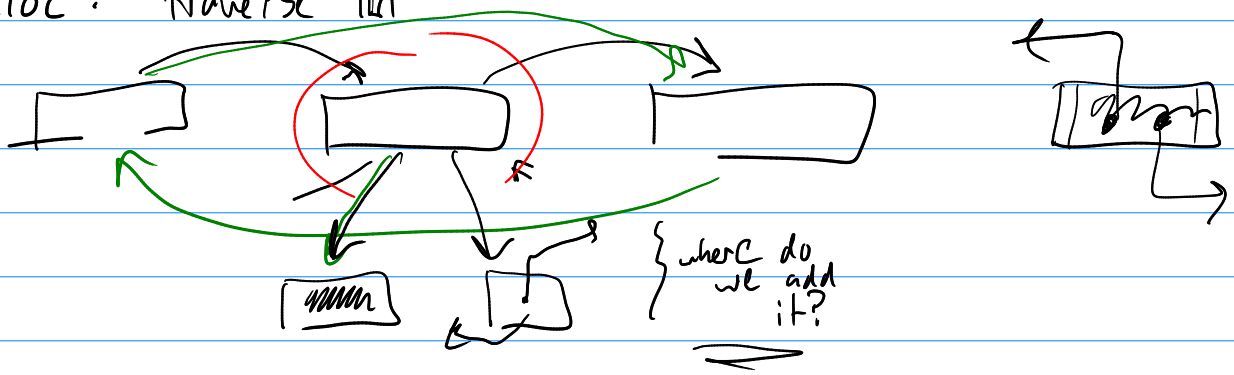
"only go through free blocks" : explicit free list







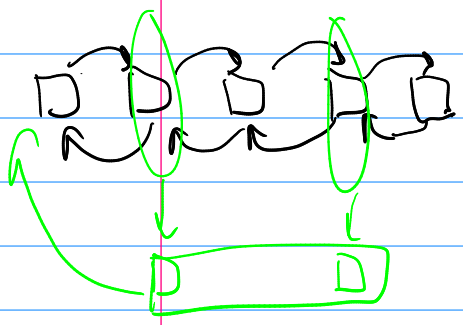
1. Alloc: traverse list

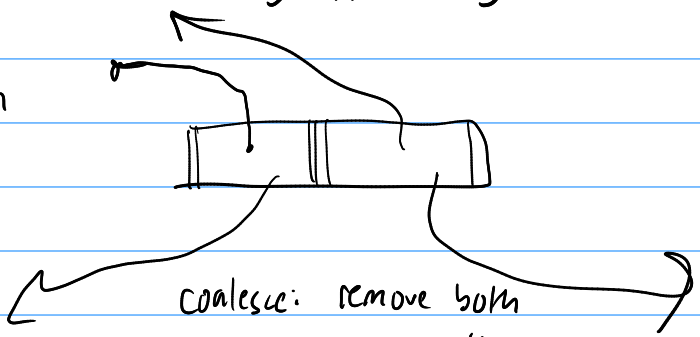


2. Free: where to insert?

LIFO: beginning of free list
- fragmentation / miss coalescing opportunity

Address ordered
- requires search

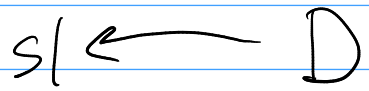




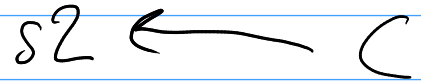
coalesce: remove both
merge mem
insert single, bigger block

Compared to implicit list
+ faster allocation
- slightly more complex
- extra mem. (minimum block size)

1. sort blocks by size (red-black tree)
2. segregated lists — different free lists
   - different heap for different size obj

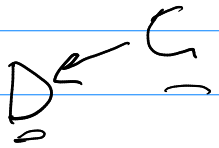


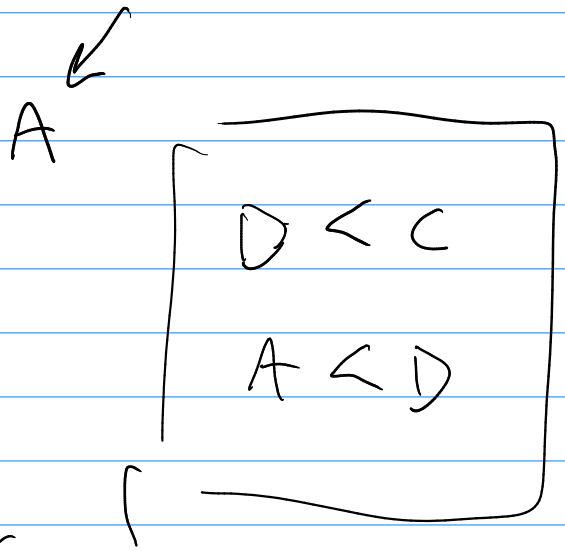
tls 0 42
tls 1 43
tls 2 44
tls 3 45
} any order



tls 4 0

174





ABCD
ACBD
ADCB

A D C

BADC
ABDC

ADBC
ADCB