

assume 1000 bytes
of heap space

- x = malloc(24)
- y = malloc(40)
- z = malloc(40)
- w = malloc(16)
- free(x) // freeing y or z would have made space for u
- u = malloc(32) // cannot use x space

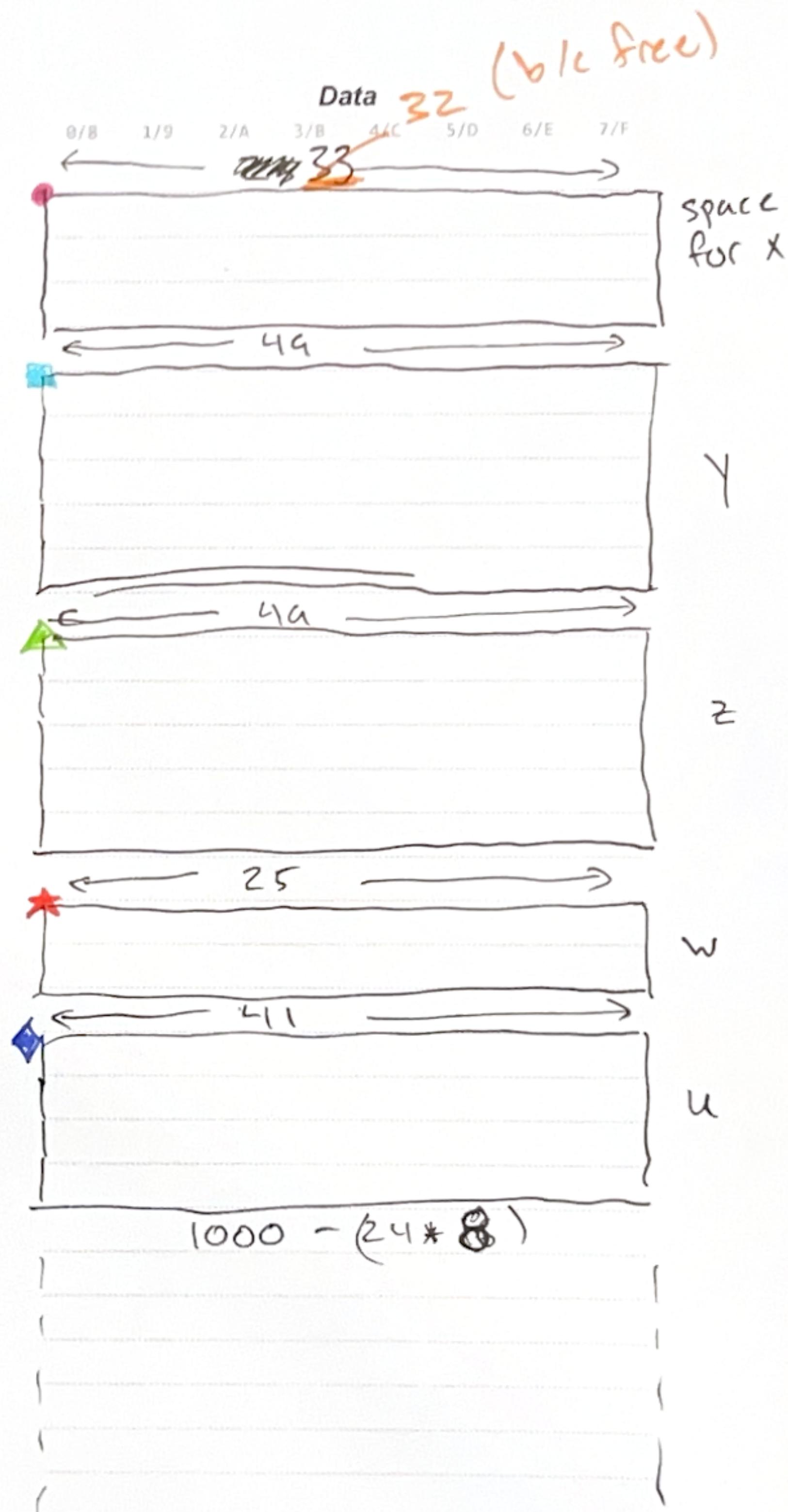
Rules for blocks:

First 8 bytes is a header

- Odd if busy/allocated
- Even if free
- Value with LSB=0 is block size
- Block size is in bytes, includes header

Lecture malloc is not PA malloc

Variable/Role	Address
HEAP_START	0x...00
	0x...08
	0x...10
	0x...18
	0x...20
	0x...28
	0x...30
	0x...38
	0x...40
	0x...48
	0x...50
	0x...58
	0x...60
	0x...68
	0x...70
	0x...78
	0x...80
	0x...88
	0x...90
	0x...98
	0x...A0
	0x...A8
	0x...B0
	0x...B8
	0x...C0
	0x...C8
	0x...D0
	0x...D8
	0x...E0
	0x...E8
	0x...F0
	0x...F8



sbrk(len)

moves "break" by
len bytes and
returns old break

