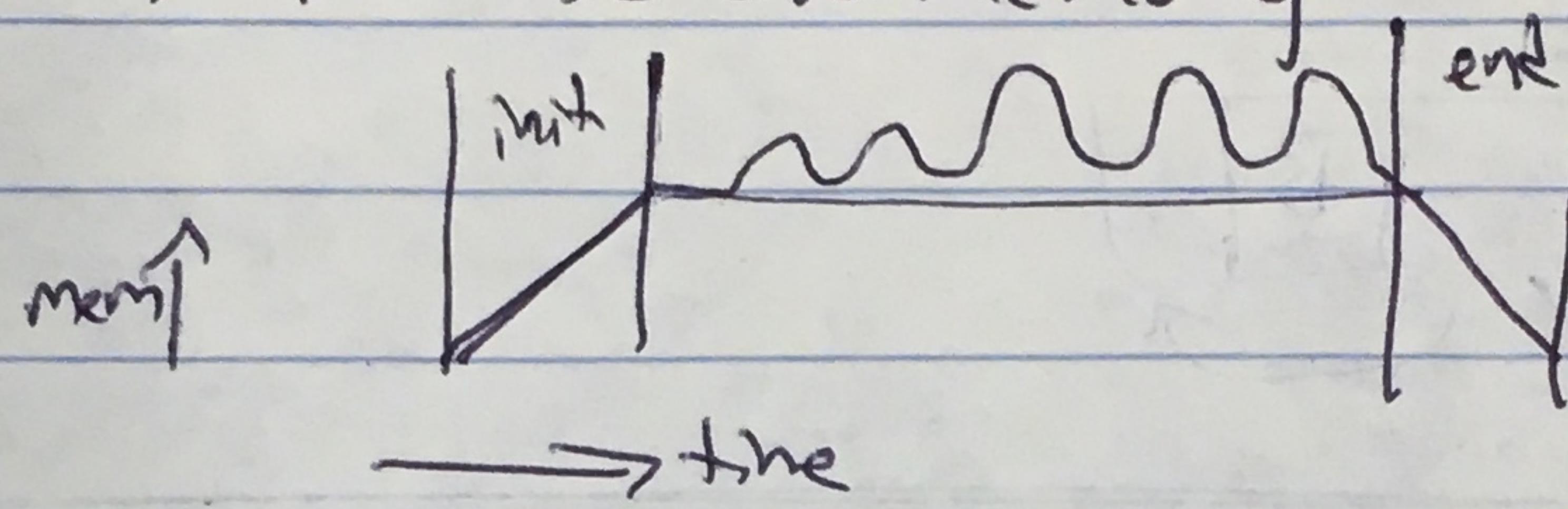


`(aStore q (numV 13)`
`8 (numV 12)`
`9 (numV 10))`

Memory → global → finite / fixed
 → linear

Finite memory

↳ reuse old memory



What can be re-used?

- if don't use it in the future → free it
- duplicated data (!) ← hash consing
- could re-calculate ← cache

`free(p)` ⇒ don't need it → humans just know

`int* f() {`

`int *p = malloc(4);`
 `return p;`

extent of values

≠

scope of variables

3

intuition = "you can't get to it"

truth

soundness

don't free
early

completeness

don't free
late

probability

`int *p = malloc(...);`

`if (f(x) == 0)`

`p = ...`

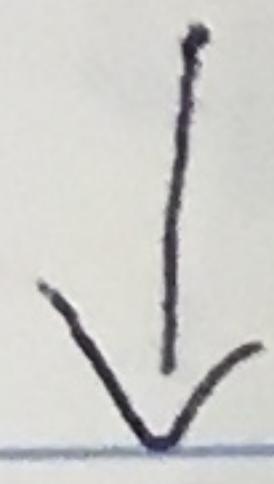
→ f = "return 0 if x
as a T.M. halts"

3 else

~~... we~~ ... we not mention p ...

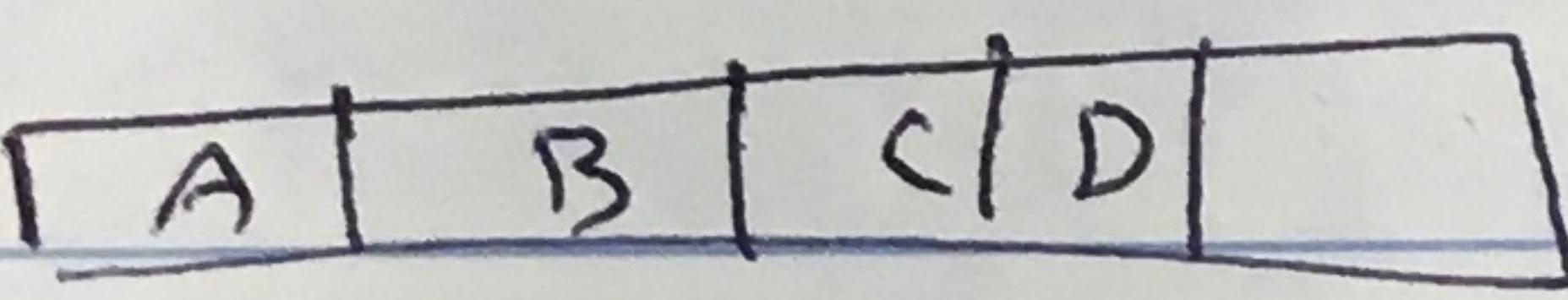
3

2

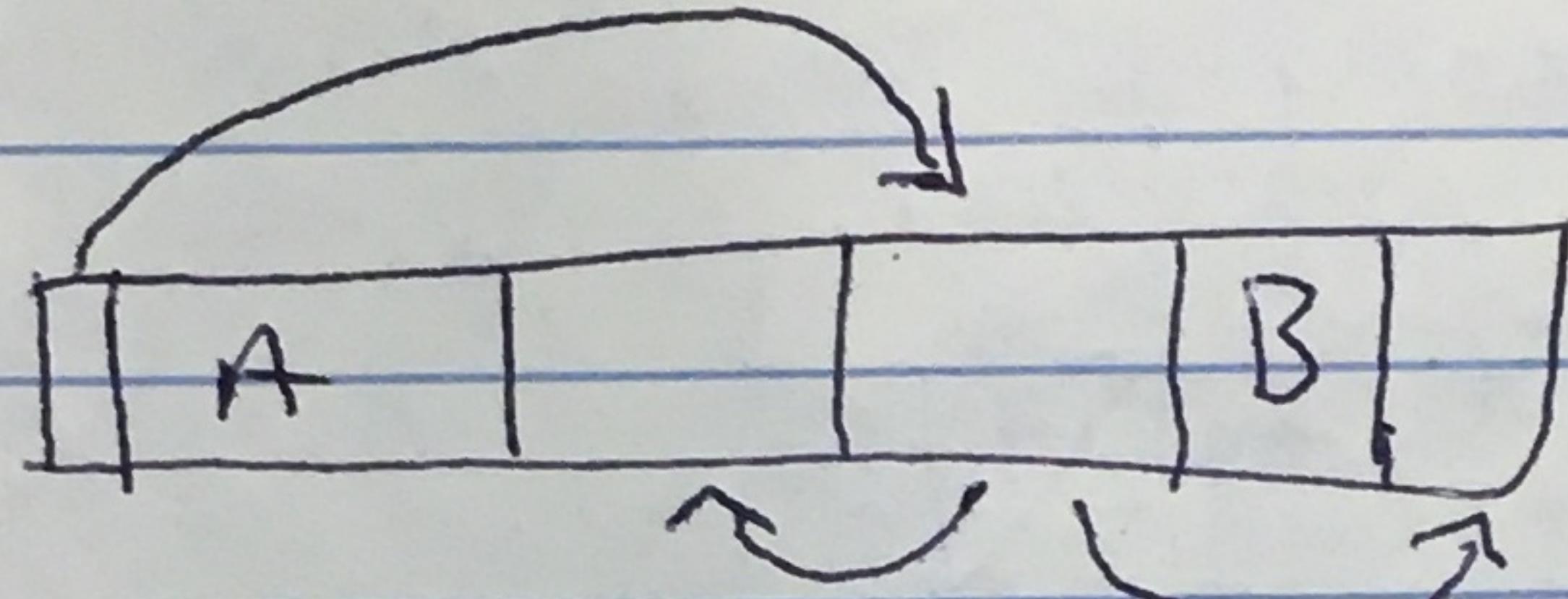
mallocideal: $O(1)$ freeideal: $O(1)$ cons

- trust people

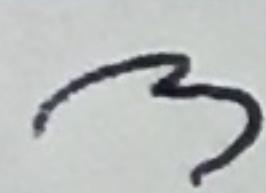
- mis how much memory

- $n \rightarrow$ # of malloc ~~$O(n)$~~ 

free(B)

 $O(\lg m)$ (some $O(\lg n)$) \rightarrow memover headideal: $O(1)$

malloc + free

tree: $O(1)$ smart pointersreference countingup() if($p \rightarrow \text{count} == \text{MAX}$) down() if max do nothing $p \rightarrow \overset{\text{size}}{\underset{\text{count}}{\text{size}}} \leftarrow \text{count} + 1$ if ($-p \rightarrow \text{count} == 0$)

free(p)

posn \star m = f(...)m \Rightarrow up()g(m) \leq am \Rightarrow down()

global = m

global \Rightarrow up()

:

malloc: same $O(\lg n)$

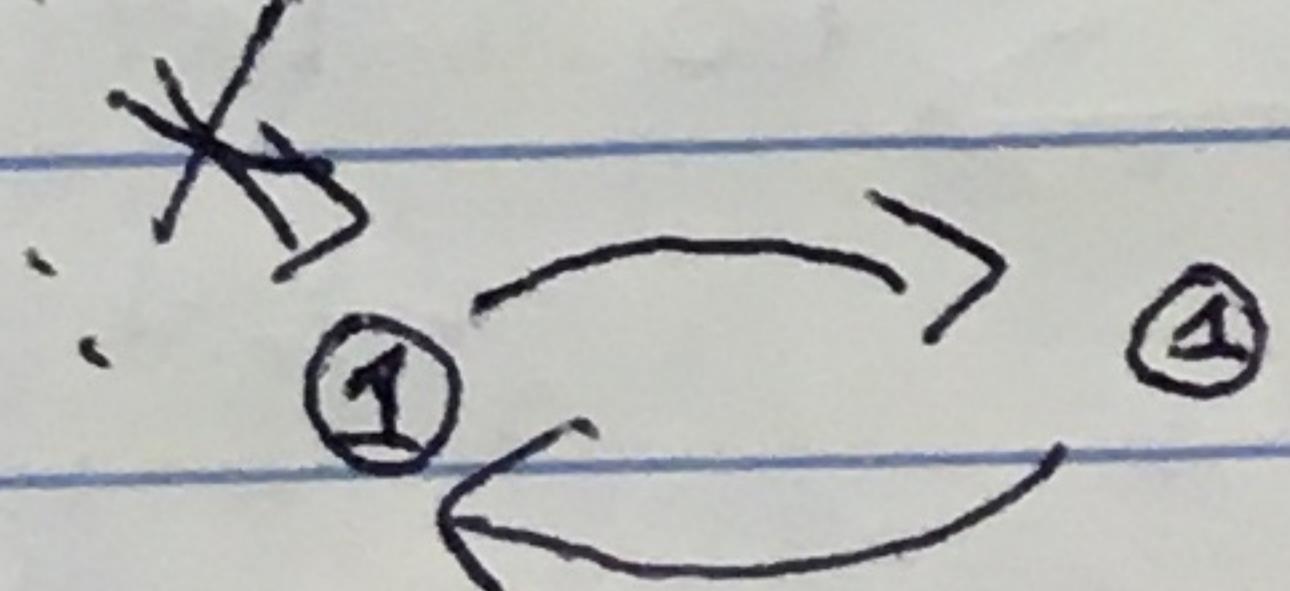
soundness: trusts humans

free: same $O(\lg n)$

how completeness:

mem: $(1 + \text{sizeof}(\text{faunt}))$ use: $O(\text{references})$

program



cyclic structures

are never freed