Memo No.
Date / /

LEC5. Linear Sorting 3.19

Problem Session 3 3.20

Problem 3-2: (reduction)

Set (Hashing)	→ Sequence
- build $O(n)$ exp. ↔	- build $O(n)$ exp.
- find $O(1)$ exp ↔	set_at and get at $O(1)$
- delete / insert $O(1)$ exp. am.	- insert/delete $\boxed{O(n)}$ exp
	- insert / delete first / last $O(1)$ e.a.

Idea: 1. Index $\overset{\text{in sequence}}{}$ =sign key to each item

get_at(i): find(i).$\overset{\text{value}}{}$, seq-build(A): set-build(<key=i, value=A[i]> for i=0, ..., |A|-1

set_at(i): find(i). value = X

insert/delete: iterate all items, insert/delete one rebuild OR use delete/insert of Hashing to shifting sequence

- insert / delete first & last:

insert_last: insert (<key len(), value X>)

insert-first: need shift all items ⇒ not O(1) time

it's O(n) X

Mo | Tu | We | Th | Fr | Sa | Su

Memo No.

Date

InvarimT: keys $= \{$ first, first+1, ⋯ first+len−1 $\}$

Idea: store variable first = key of first item (index i)

$\Rightarrow$ Insert_first: decrement first key to −1, −2 ⋯,
insert (key first, value x)

delete_first: increase first key
delete (key first −1,) ~~ve~~

all of interfaces of before should plus first to i which inital first is 0

Problem 3.-3: ~~Cix~~ Critter sort

Sort n objects by keys

(a) an integer x_i between −n and n. $[-n, n]$ $\overset{+n}{\in} [0, 2n]$

Radix sorts n ints $\in \{0, \cdots, u-1\}$ in

$O(\overset{n}{\cancel{x}} + n \log_n u)$.–

"$O(n)$ is $u = n^{O(1)}$

(b) strings over 26 letters of len $\leq 10 \lceil \lg n \rceil$

tuple sort $\equiv$ radix sort on base 26

(c) integer f_i under i^2

(d) 1. $O(n \lg n)$ via merge sort

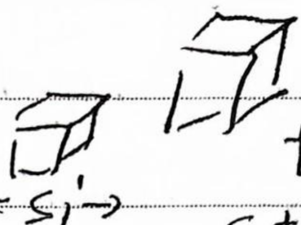
Memo No.

Date / /

(d) in $O(n)$ time

Problem 3-4.

$S = \{ S_0, S_1, \cdots S_{n-1} \}$



want 2 numbers $\in S$, summing to h

(a) $O(n)$ expected time

~~ted~~ idea 2.: build hash table on S

idea: call find() $O(n)$ times.

idea: loop over S

for $S_i \in S$: find $s_j \in S$ such that $S_i + S_j = h$

$\equiv$ find $(h - S_i) \Rightarrow O(1)$ exp

so $O(n)$ exp

(b) $O(n)$ worst-case

find biggest peir wise sum $\leq h$, no $S_1 + S_2 = h$

assume $h = 600n^6$ find S_1, S_2 close to h

Idea 1: radix sort .. but S_i colud be $>> h$

$\Rightarrow$ Idea: $S_i > h \Rightarrow$ throw away $\Leftarrow$ radix sort

$\rightarrow S_i$ is sorted. for $S_i \in S$, $= 0, 1, \cdots, n-1$

S_i binary search) for every $S_i \Rightarrow O(n \cdot \log n)$

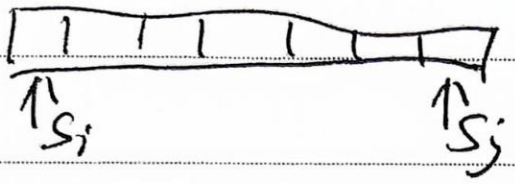
$h - S_i$, bs can tell me most approach number

amotization amortization ~~amorit~~
amortizatione amortization
amortization

Memo No.
Date / /

⇒ return lagest $\overset{c}{\cancel{d}}$ candidate

Idea:



$\boxed{\text{two-finger Algorithm}}$

$S_i + S_j$ ⇒ how close to h

① $S_i + S_j > h$ ⇒ $j = j - 1$ decrement j

② $S_i + S_j < h$ ⇒ increment i
↳ add to candidate

Invariant $S[i'] + S[j'] \overset{>h \text{ or}}{\leq h}$ candidate'
for all $i' \leq i \leq j \leq j'$

$i \geq j$ stop return max candidate

P3-5 cut_i $\overset{\downarrow i=2}{ab|cdbc}$ ← D

cdbc ab

$\underbrace{\text{cdbc}}$ $k=4$ → deal k cdbc $P(D, i, k)$

$\xrightarrow{\text{sort}}$ bccd

① $P(D, i, K)$

26 # ∈ [0, n]
↳ base n+1, 26 digits $\boxed{(n+1)^{26}}$

aaba
bu aa
ab aa
} all same aaaab