



01062023
Extra Class

What is loop invariant

feasibility

formal \rightarrow revision

a lot application 73

200

AIM!

(23.0)

22.5

recurrence relationship \rightarrow master theorem

loop invariant? depends

avl ✓

bipartite ✓

suffix ✓

dynamic X

feasibility? depends

② algorithm
data structure

working sheet Only

X partial martingale

① correctness
complexity
(loop invariant)

③ application
TAT (no mCQ)

time, space complexity (Worst case complexity)

Output sensitive algorithm? not yet

↳ complexity depends on output

recurrence relationship (MCQ)

↳ find time complexity

↳ master theorem

↓ variable (which is main variable to count n)
complexity

↳ relate to base case

↳ getting smaller

prove algorithm is correct

↳ termination, loop invariant

↳ ask individually ↳ more important

- ① initialisation
 - ② maintenance
 - ③ termination
- } explain / argue / proof
common sense (highlight)

sorting
→ comparison
→ non-comparison

} come out BUT
less marks

quiz
→ loop invariant
→ sorting
divide & conquer

difficult sorting

↳ radix is constant w.r.t
bigger base

bigger input/item = expensive comparison
↳ more item to compare

explain why each of
them are stable /
in-place or not

n^2 biggest num in radix → complexity → linear

quicksort } counting sort / radix sort

- ① complexity
- ② manipulate base to achieve complexity

← quicksort → $O(n \log n)$
quickselct → $O(n)$

} divide &
conquer

V in complexity = graph

improve quicksort worst comp
↳ use median of medians

improve partitioning
dutch national flag (dnf)
↳ improve best case

* t = pivot partition

loop invariant

radix linear for numbers ALWAYS!
↳ manipulate base for always linear

↳ pseudocode start 1

↳ python start 0

↳ follow question if its 1..n or 0..n

dynamic programming

- * top-down = bottom-up
 - (recursion)
 - (iteration)
- * backtracking can be examinable !
- ** no coding question in exam

factor affecting time complexity

- ↳ memory
- ↳ time taken to decide what to fill up
- ↳ number of spaces to fill up

Ian advise → ① mcQ ? tembak

- ② subjective ? write something
- ③ skip Imao

Hashtable

↳ store 2 key item ↳

hash table best case $O(1)$

HENCE, always act worst case

* unique key

hash function hash key to unique place in hashtable

linear probe

* quadratic

let everyone

have individual

linear vs quadratic

which is better?

↳ got collision?

space but

don't use

all space

* linear don't waste empty
space but cause primary
clustering

point of collision + 1

collide again? point of collision + 2

quadratic probe

→ can cause primary and secondary
clustering but answer
secondary clustering

↳ got collision?

point of collision²

first

linear probe

collide again? point of collision⁴

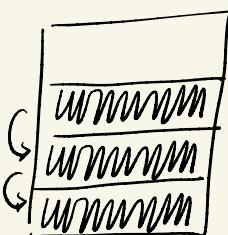
→ side by side

clustering

→ primary: cluster between hash value

→ secondary: cluster between same hash

→ side space side



hash here every time

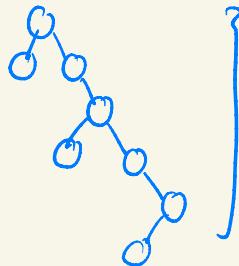
double
hashing

"no space next space"

AVL "self balancing tree"

don't mess up or ian write referral letter to sunway

$$\text{balance factor} = \{-1, 0, 1\}$$

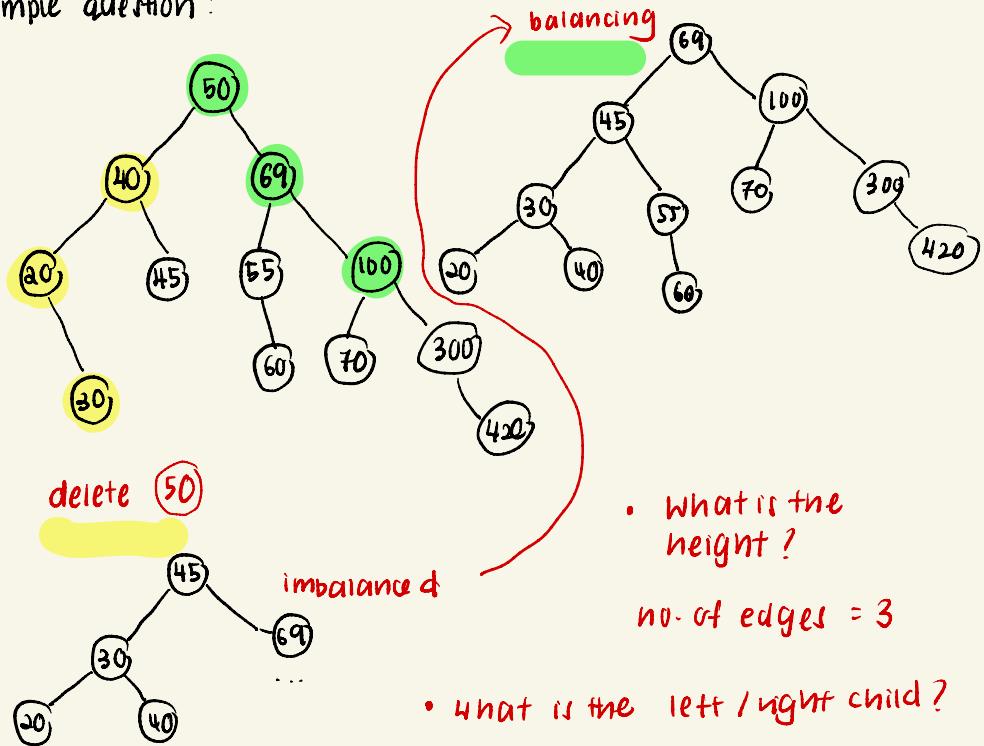


not possible
to come out

11
when 2 num
the same
follow the
previous one "

LL ... L | RR ... R

example question:



suffix tree and tree

- string matching
- examinable:
 - x pseudocode
 - ✓ complexity \rightarrow $m \times m$ number of nodes (max)
 - \rightarrow height m (worst)
 - x draw

suffix vs normal tree

Application question? high chance no

... unless ... new lecturer evil think

suffix - array \rightarrow in order

\hookrightarrow sort suffix by suffix ID

* index start 1, not 0 but
not really matter

smallest lexicographic? $\$ \rightarrow$ ~~ALDZ~~
 \rightarrow ~~ALDZ~~ \rightarrow ~~ALDZ~~
then

prefix doubling O(1)

rank [$i+k$] \rightarrow at step $2k$

sample question

q18

$t=2$

$2t=4$

compare 5 & 7

$$\begin{array}{r}
 \begin{array}{cc}
 105 & 107 \\
 2 & 2 \\
 +t & +t \\
 \hline
 107 & 109
 \end{array}
 \quad
 \begin{array}{cc}
 103 & 105 \\
 2 & 2 \\
 +t & +t \\
 \hline
 105 & 107
 \end{array}
 \quad
 \left. \begin{array}{c} \\ \\ \end{array} \right\}
 \end{array}$$

rank 3 = 2
rank 5 = 2

when rank 3 = 2 + 2,
rank 5 = 2

when rank 5 = 2 + 2,
rank 7 = 2

both ranks the
same so it can't
be resolved at $2t=4$

Step ① compare

↳ same? go step ②

↳ diff? resolve

Step ② compare

↳ diff? resolve

↳ same? not coincident

Half exam = graph

↳ include flow network

minimum spanning tree

directed acyclic graph

adjency matrix

· dijkstra $e \log(v)$

adjency list

· bellman ford

whatif matrix \leftrightarrow list?

→ wide

bfs vs dfs → depth

} $O(V+E)$

· tracing bfs & dfs is difficult

100% come out:

① dijkstra

② bellman-ford

③ floyd-warshall

negative cycle
how to find

loop one more time

intermediate

k i j

↳ need matrix

dragon

graph can be disconnected until explicitly stated in question

↳ sanity check has explained

terminate earlier when diagonal is negative

prim and kruskal

↳ no union find!

complexity

prim vs dijkstra

↓
look at
edge
only

↓
look at
source...

given a graph,

build a minimum spanning tree

If disconnected,
work out the max
disconnected component

how to know if
it is unique?

all weight different?
all weight same?

can modify to find
max spanning tree

→ prim: use max heap

→ kruskal: sort other way around

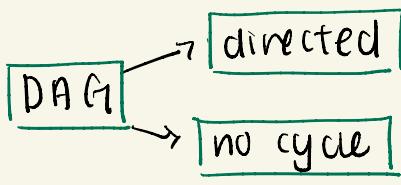
negative edge?

tree no cycle

bpm? don't care, smaller is good

Kruskal? also don't care, sorted monotonically,
smaller is good

MST diff from shortest distance?



→ bfs
→ $O(V+E)$

kahn = x unique

when vertex insert queue?
no incoming edge

unique or not unique
depends on size of queue

Application in tutorials

network flow

IMPORTANT

- * flow / capacity one number = capacity
- * residual network 2 number

lower bound / flow / capacity

max flow network = ford-fulkerson method
no path = min cut exist

capacity \times same = not feasible

how to know

- max flow
- min-cut

every vertex cannot reach is the sink

bipartite matching | application for | how network

GDS

circulation \rightarrow X source, X sink

bipartite matching \rightarrow V source, V sink

feasible network \rightarrow water cycle

model flow network for feasibility \rightarrow X source, X sink

application ?

power plant

road from A to B but must ensure
the road must be used

How to know if its a feasible question

is it possible for... to make...

? S \$ S U G C G U

— TH — R Y T G U T — I