

Maximum

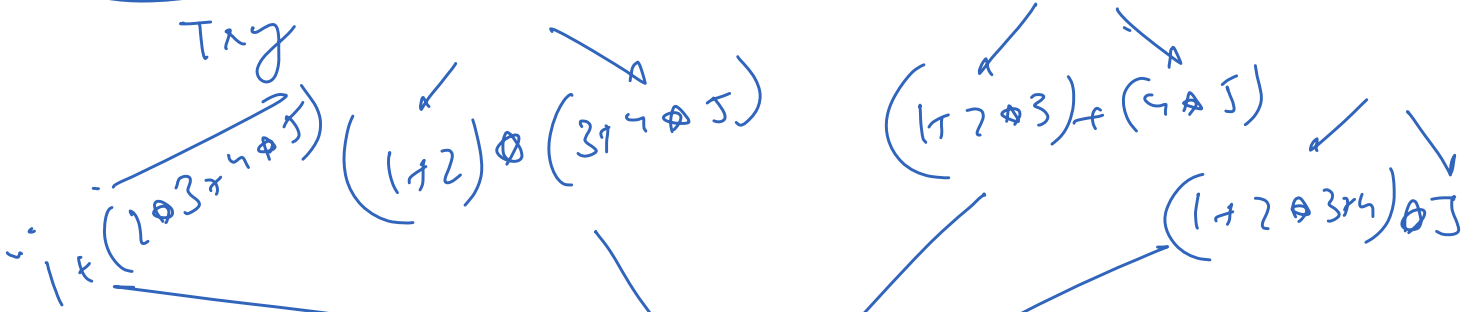
opnds [1, 2, 3, 4, 5]

expr = "1+2*3+4*5" →

oprm [+, *, +, *]

9:08-9:15

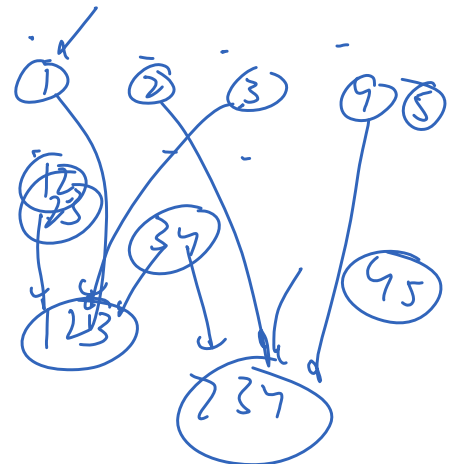
Try



1 + 2 * 3 + 4 * 5

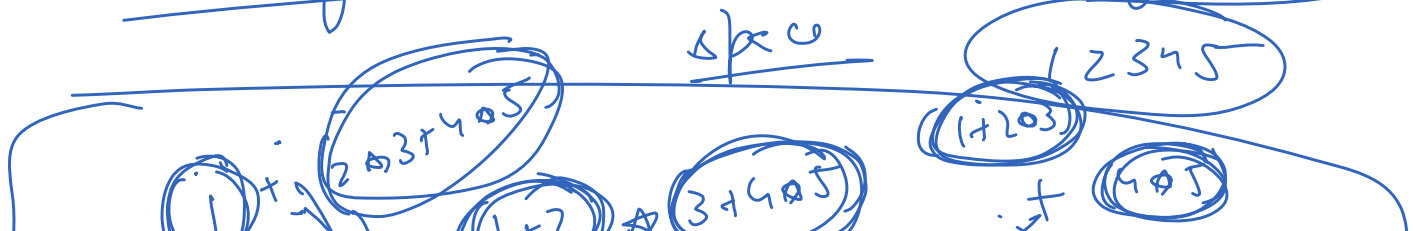
	1	2	3	4	5
1	1	3	9	21	105
2	X	2	6	14	70
3	X	X	3	7	35
4	X	X	X	4	20
5	X	X	X	X	5

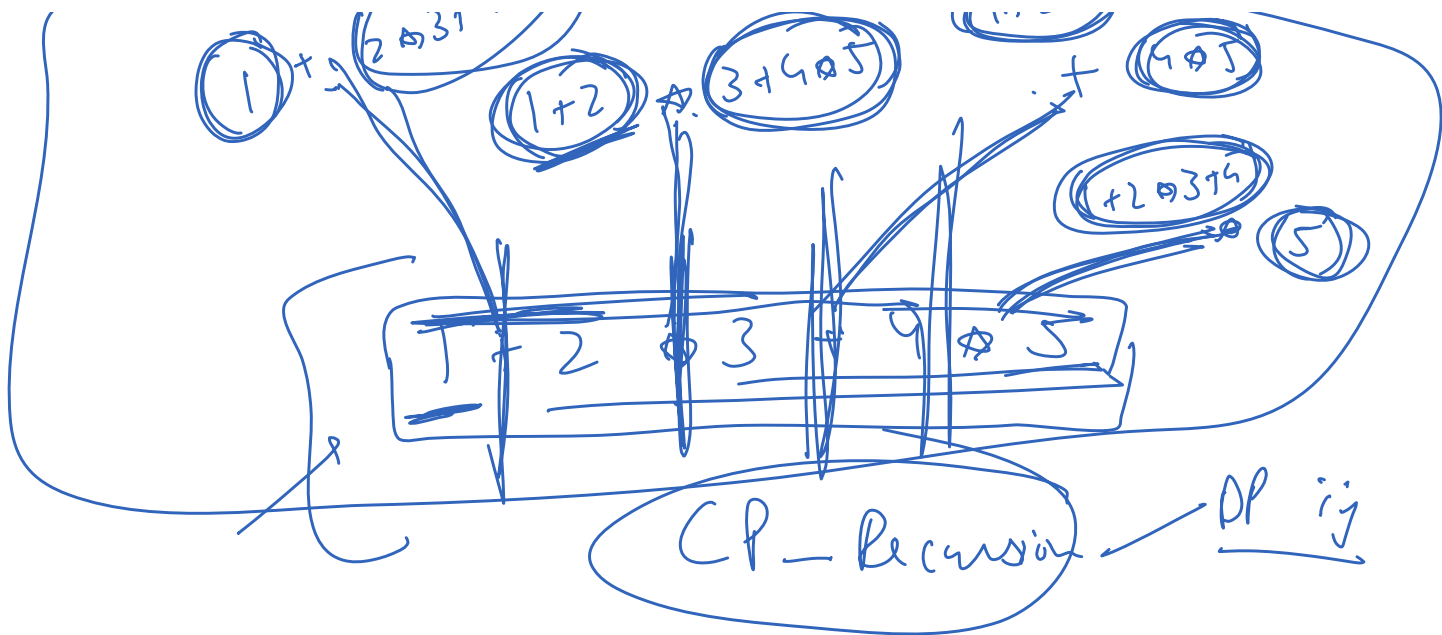
TP



Recursion,

DP defines Topological sort of solutions





Optimal binary search tree

Hard

Accuracy: 72.18%

Submissions: 5118

Points: 8

9:51 - 9:57

Given a sorted array **keys**[0.. n-1] of search keys and an array **freq**[0.. n-1] of frequency counts, where **freq**[i] is the number of searches to keys[i]. Construct a binary search tree of all keys such that the total cost of all the searches is as small as possible.

Let us first define the cost of a BST. The cost of a BST node is level of that node multiplied by its frequency. Level of root is 1.

Example 1:

Input:

n = 2

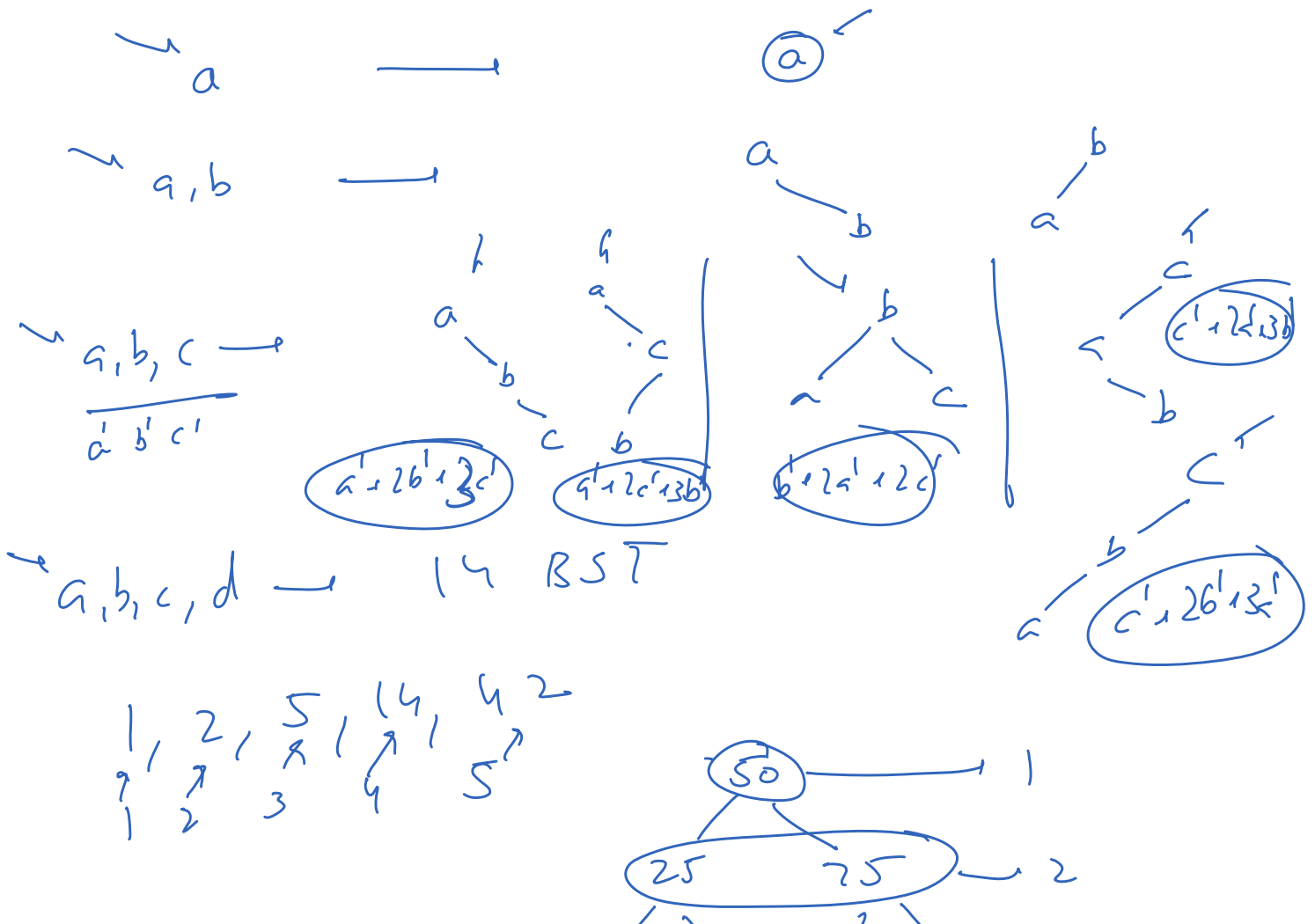
keys = {10, 12}

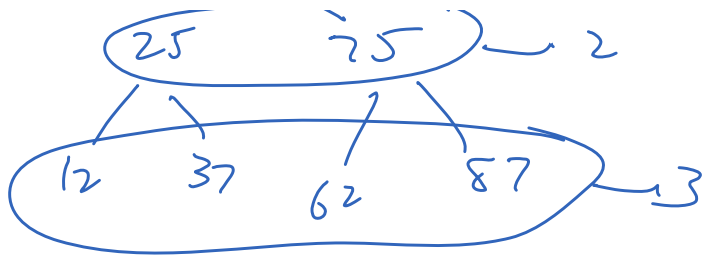
freq = {34, 50}

Output: 118

Explanation:

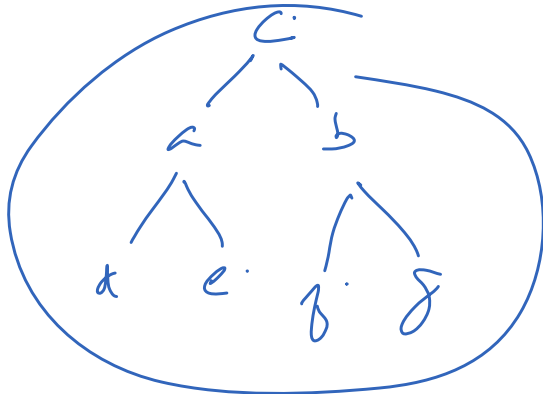
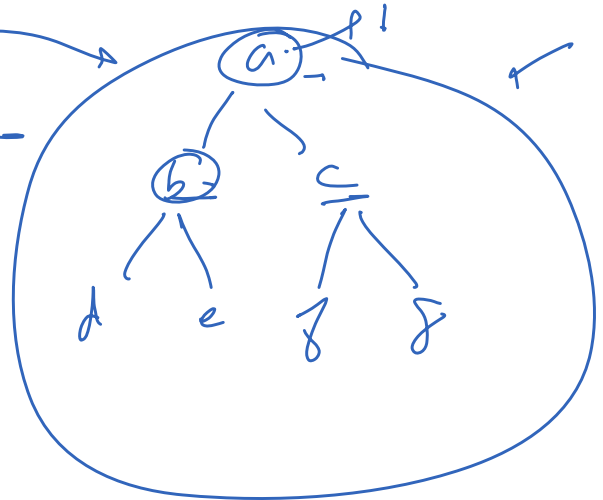
There can be following two possible BSTs





$v \rightarrow a, b, c, d, e, f, g$

$f \rightarrow$	a'	b'	d'	d'	e'	f'	g'
$c \rightarrow$	1	2	2	3	3	3	3



$$\begin{aligned} & 1.c' + 2.d' + 2.b' \\ & + 3.d' + 3.e' + 3.f' + 3.g' \end{aligned}$$

$$\begin{aligned} & \underline{1.a'} + \underline{2.b'} + \underline{3.c'} + \underline{4.d'} + \underline{5.e'} \\ & + \underline{6.f'} + \underline{7.g'} \end{aligned}$$

$a-b-c-d-e-f-g$

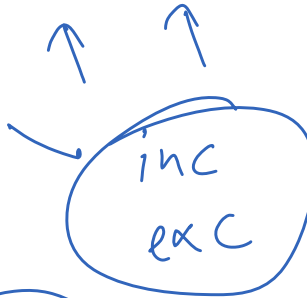
Optimal BST

Cost

Greedy Value of Decision
Proof

proof

77



200

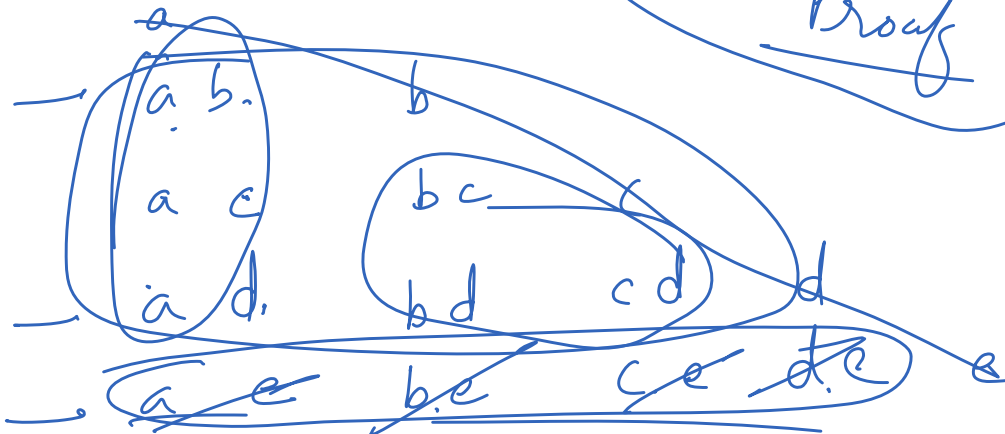
Greedy
proof

a
a b
a b c
a b c d
e

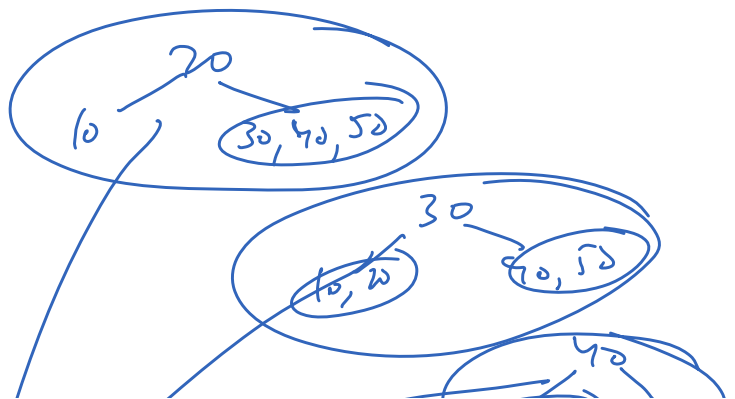
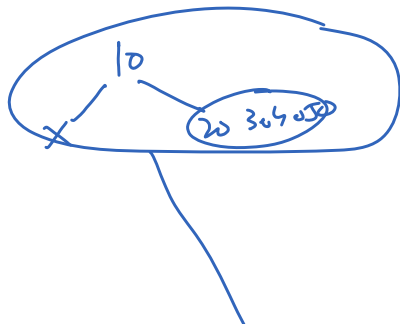
a b c d e

$a + e > t$

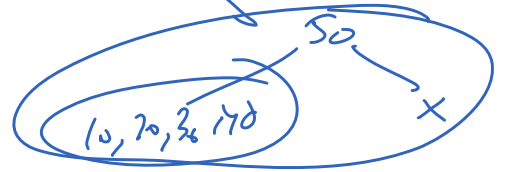
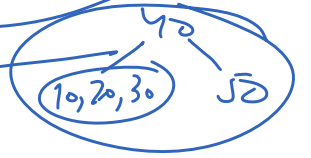
$a + d < t$



$v \rightarrow 10, 20, 30, 40, 50$
 $q \rightarrow 2 \quad 3 \quad 1 \quad 5 \quad 4$

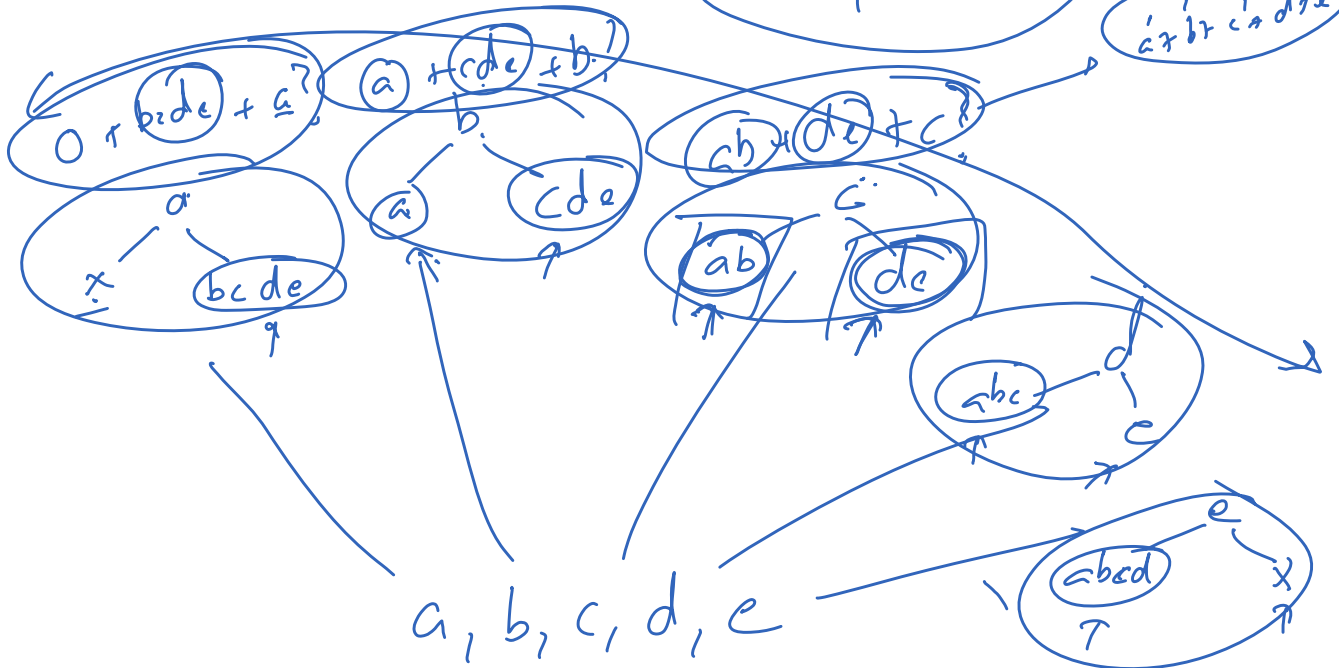


10, 20, 30, 40, 50



10:34 - 10:39

a' b' c' d' e'



$a', b', c', d' + a' + b' + c'$

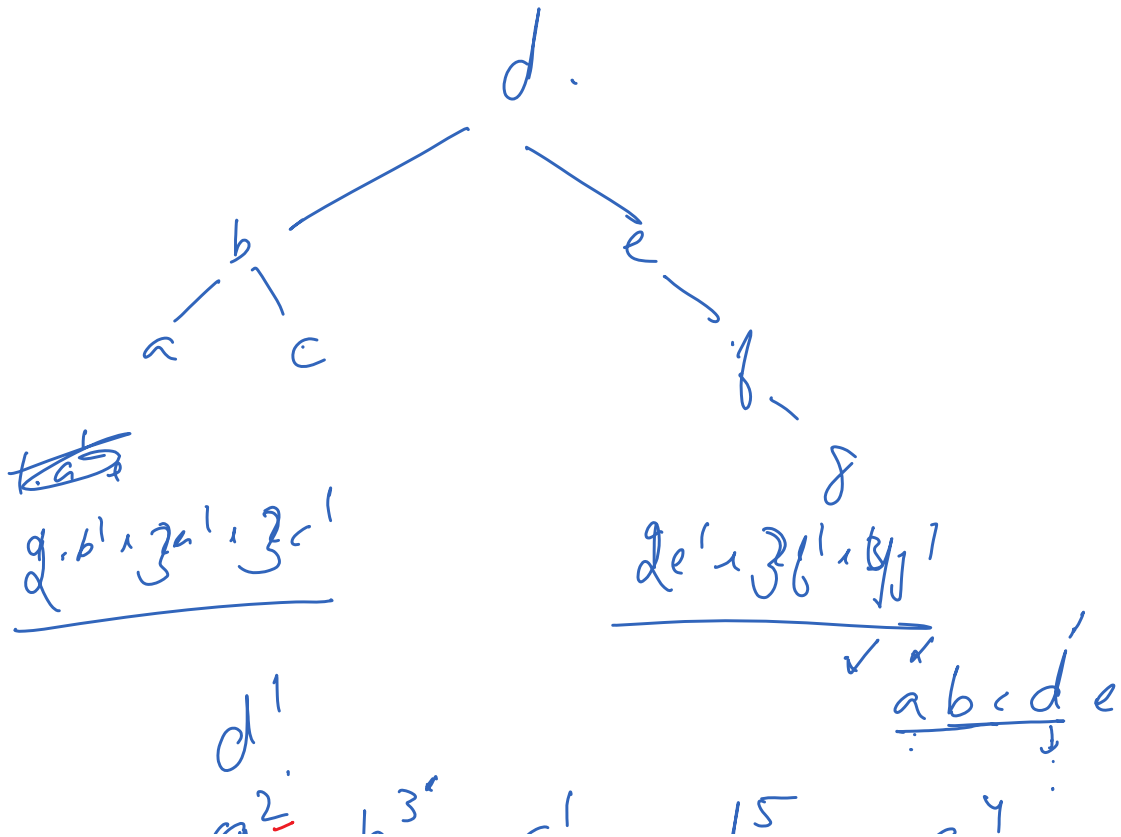
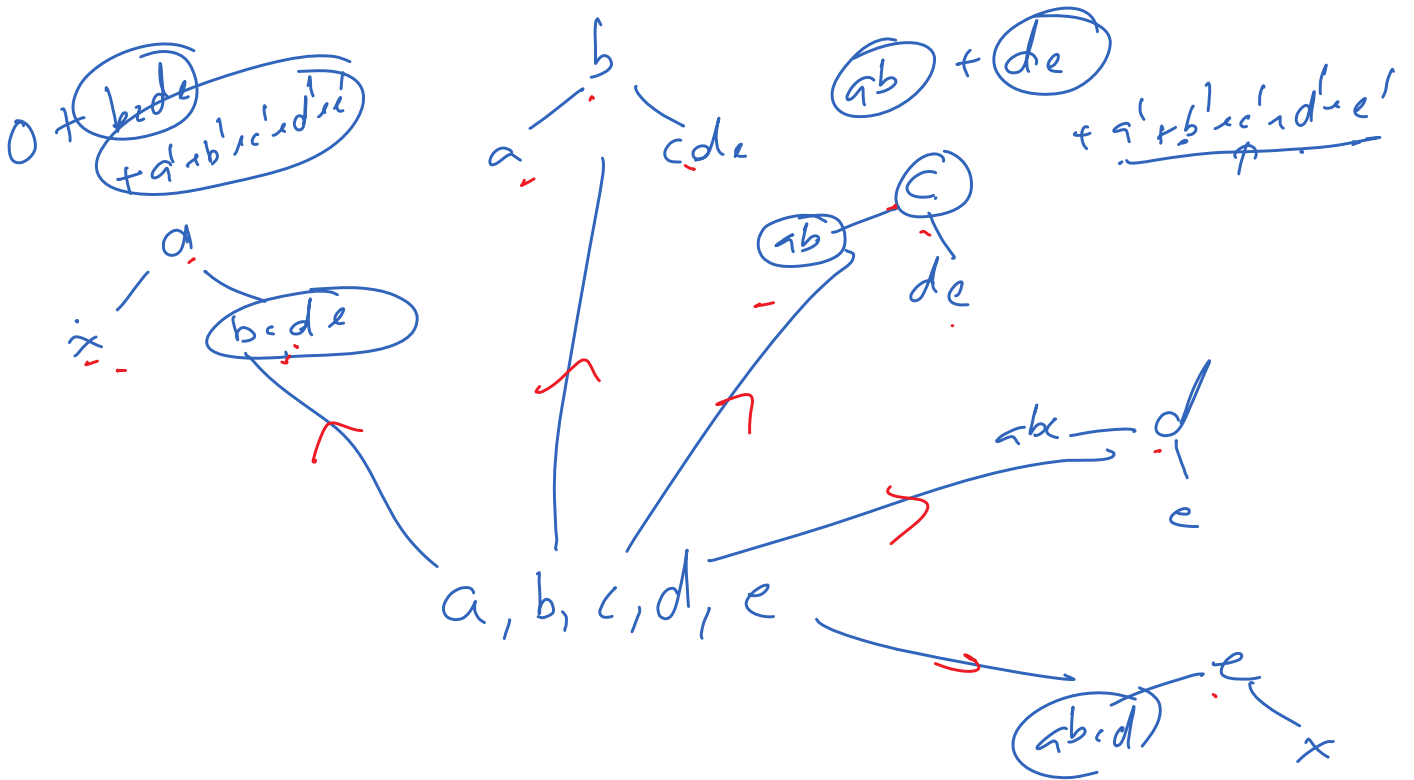


$1 \cdot b' + 2 \cdot a' + 2 \cdot c'$

$1 \cdot g' + 2 \cdot e' + 2 \cdot f'$

$1 \cdot b' + 3 \cdot a' + 3 \cdot c' + 2 \cdot f' + 3 \cdot e' + 3 \cdot g'$

$$\underline{1, d'} \quad \downarrow \quad 2, b' \quad \downarrow \quad 3, a' \quad \downarrow \quad 3, c' \quad \downarrow \quad 2, f' \quad \downarrow \quad 3, e' \quad \downarrow \quad 3, g'$$

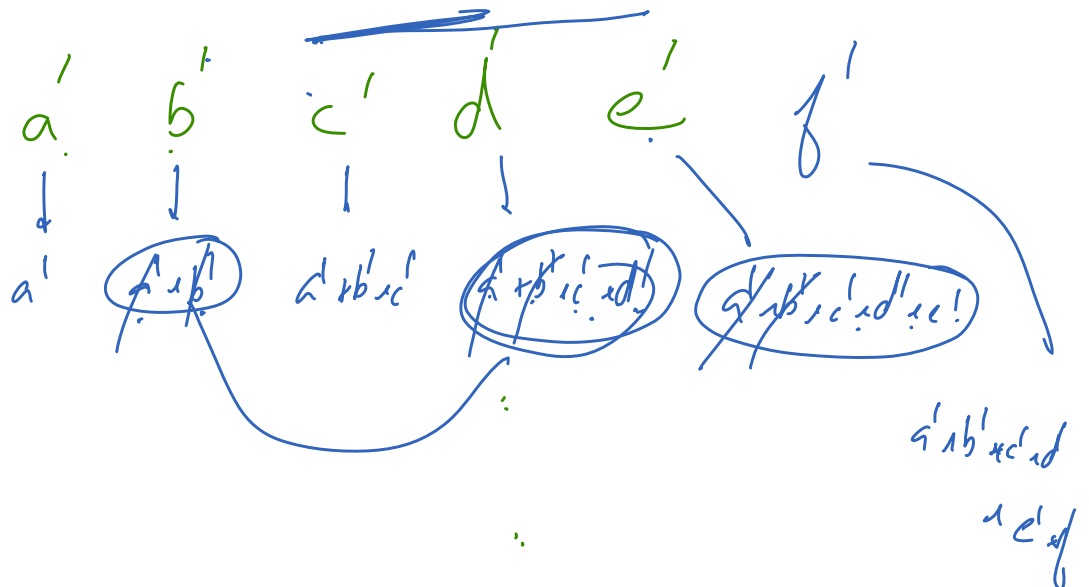
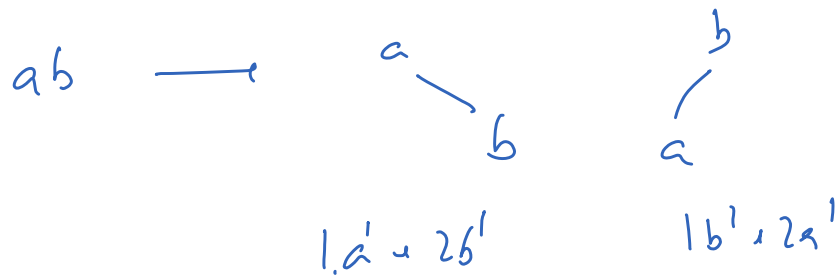


	a^2	b^3	c^1	d^5	e^4	
$i, j=5$	<u>a</u>	<u>ab</u>	<u>abc</u>	<u>abcd</u>	<u>abcde</u>	i, j
b^3	X	3	5		<u>bcd</u>	$i+1, j$
c^1	X	a	c	7	<u>cde</u>	$i+2, j$
d^5	X	a	a	5	<u>de</u>	$i+3, j$
e^4	X	a	a	a	4	$i+4, j$
						$i+5, j$

MCM

Min-Max

Optimal BST



316. Remove Duplicate Letters

Medium

5897

380

♥ Add to List

📄 Share

Given a string s , remove duplicate letters so that every letter appears once and only once. You must make sure your result is the **smallest in lexicographical order** among all possible results.

Example 1:

Input: $s = "bcabc"$

Output: $"abc"$

Example 2:

Input: $s = "cbacdcbc"$

Output: $"acdb"$

Constraints:

What?

How?

Why?

~~b~~
~~d~~
c
a

~~f~~ ~~b~~ ~~a~~ ~~c~~ ~~d~~ ~~c~~ ~~b~~ ~~c~~

$c \rightarrow \text{not } 0$

$b \rightarrow \text{not } 0$

$a \rightarrow \text{not } 0$

$d \rightarrow \text{not } 0$

$acdb$

~~b~~ ~~d~~ c a

$acdb$

~~dup X~~

(dup X)

atleast one

Monotonic stack

dup

atleast once

~~a~~ ~~b~~ ~~c~~ ~~d~~ = ~~a~~ ~~b~~ ~~c~~
~~a~~ ~~b~~ ~~a~~ ~~c~~ = ~~b~~ ~~a~~ ~~c~~
~~a~~ ~~b~~ ~~a~~ ~~c~~ = ~~a~~ ~~d~~ ~~c~~ ~~b~~
~~a~~ ~~b~~ ~~a~~ ~~c~~ = ~~b~~ ~~a~~ ~~d~~ ~~c~~
~~a~~ ~~b~~ ~~a~~ ~~c~~ = a ~~c~~ ~~d~~ ~~b~~
~~a~~ ~~b~~ ~~a~~ ~~c~~ = ~~b~~ ~~a~~ ~~c~~ ~~d~~
~~a~~ ~~b~~ ~~a~~ ~~c~~ = ~~c~~ ~~a~~ ~~d~~ ~~b~~
~~a~~ ~~b~~ ~~a~~ ~~c~~ = ~~c~~ ~~b~~ ~~a~~ ~~d~~

a b c d

dup

rem

11:53-12:03

402. Remove K Digits

Medium 6611 277 Add to List Share

Given string num representing a non-negative integer num, and an integer k, return the smallest possible integer after removing k digits from num.

Example 1:

Input: num = "1432219", k = 3

Output: "1219"

Explanation: Remove the three digits 4, 3, and 2 to form the new number 1219 which is

✓ ✓ ✓ ✓ ✓ ✓ ✓
1 4 3 2 2 1 9

8

Input: num = "1432219", k = 3

Output: "1219"

Explanation: Remove the three digits 4, 3, and 2 to form the new number 1219 which is the smallest.

3

Example 2:

~~XXX~~

9
1
2
~~2~~
~~4~~
~~3~~
1

1 2 3 4 5 6
6
5
4
3
2
1

✓ ✓ ✓ ✓ ✓
1 0 4 5 6

6
5
4
0
~~X~~

~~456~~

6 5 4 0 ...

~~✓✓✓~~
 1 1 2
 ✓ ✓ ✓

✓ ✓ ✓ ✓ ✓ ✓
 1 0 0 4 3 5

~~20~~ 85

~~2~~
 1
 1

5
 4 3
 0
 0
 1

a
 |
 b
 |
 c
 |
 d
 |

d
 c
 b
 a