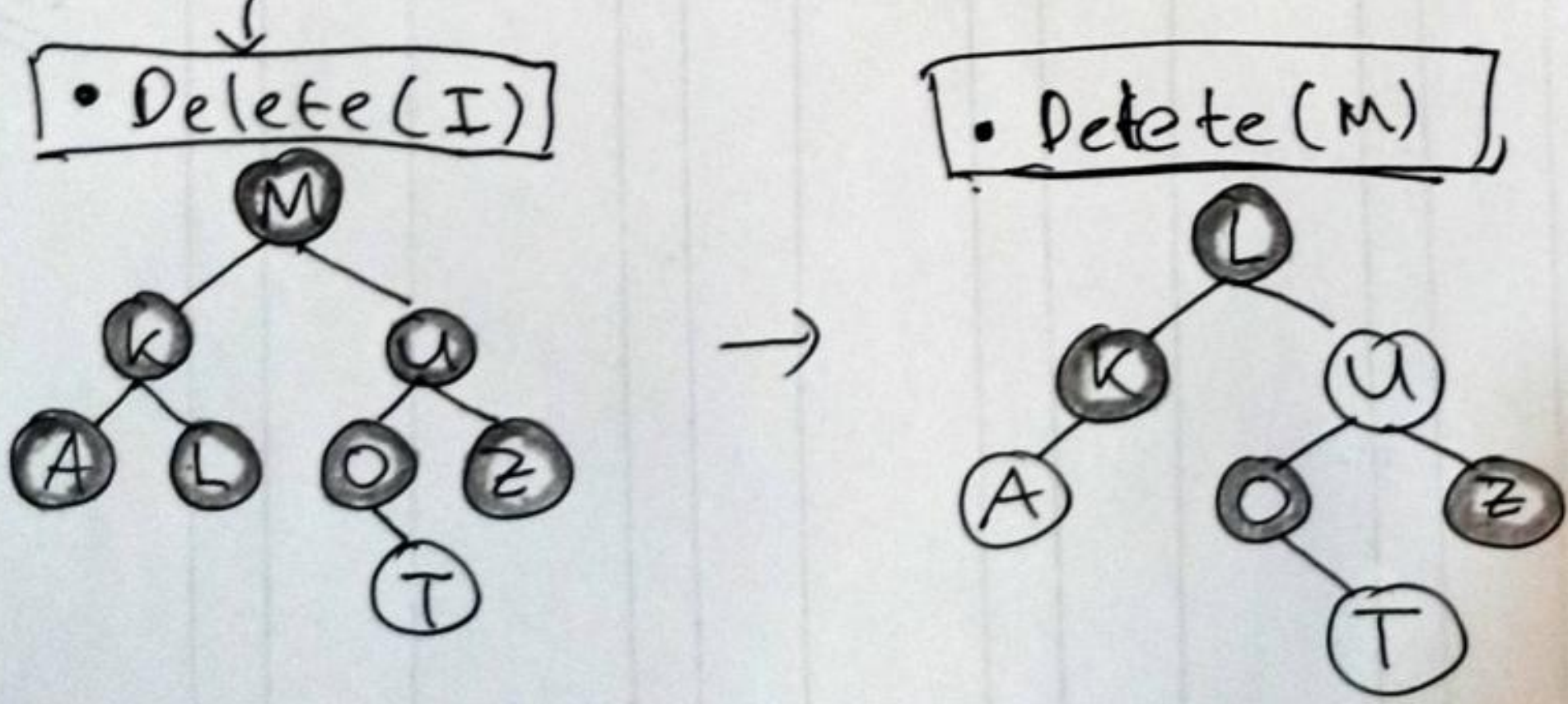
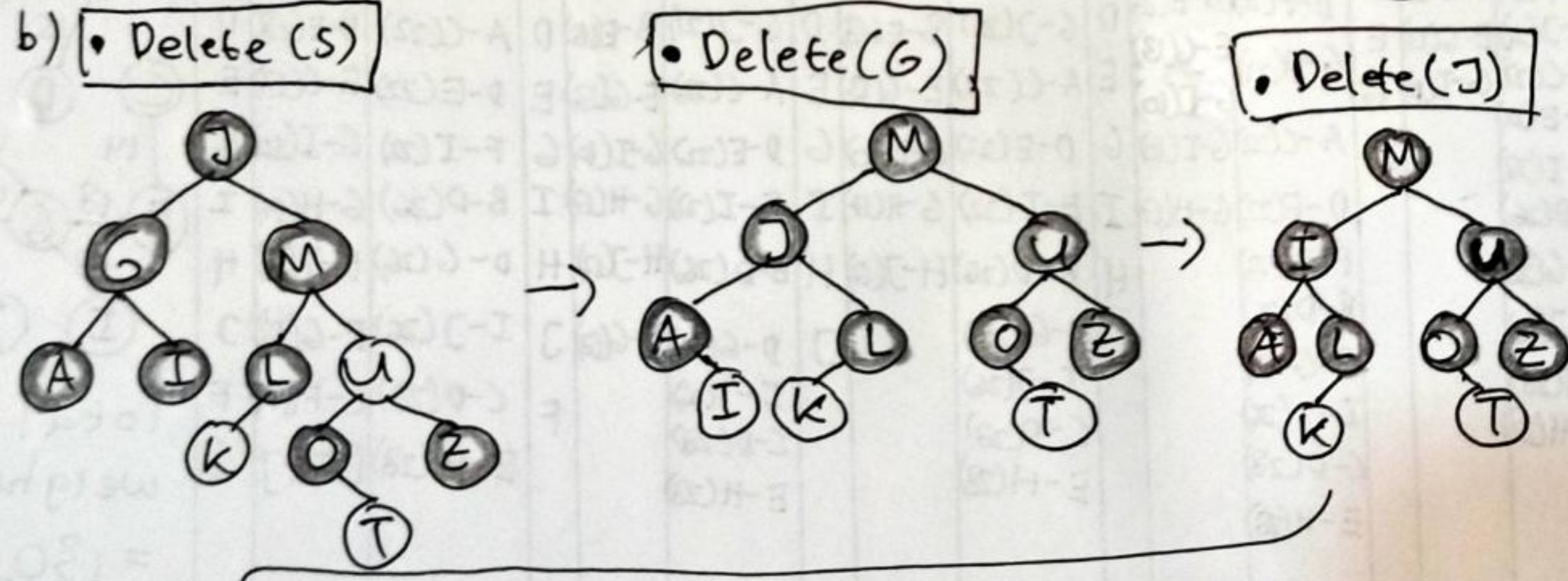
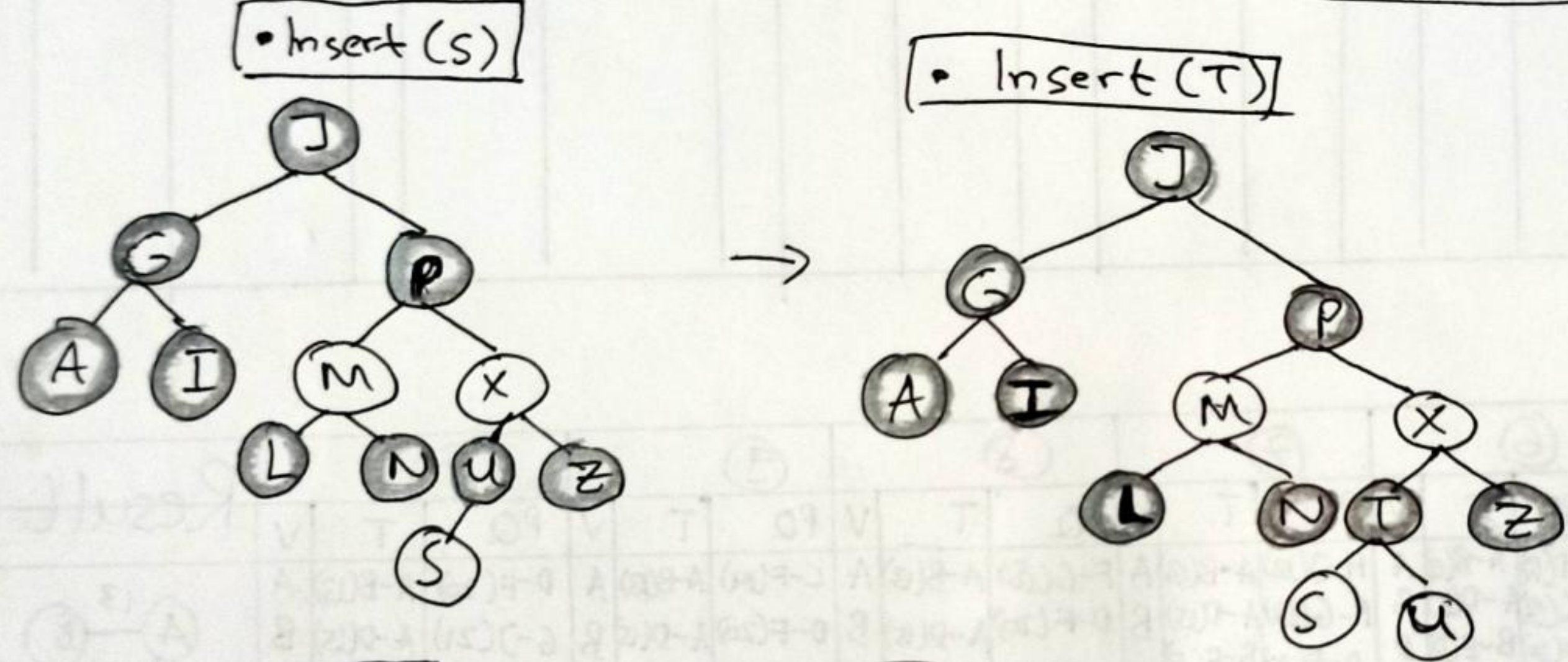
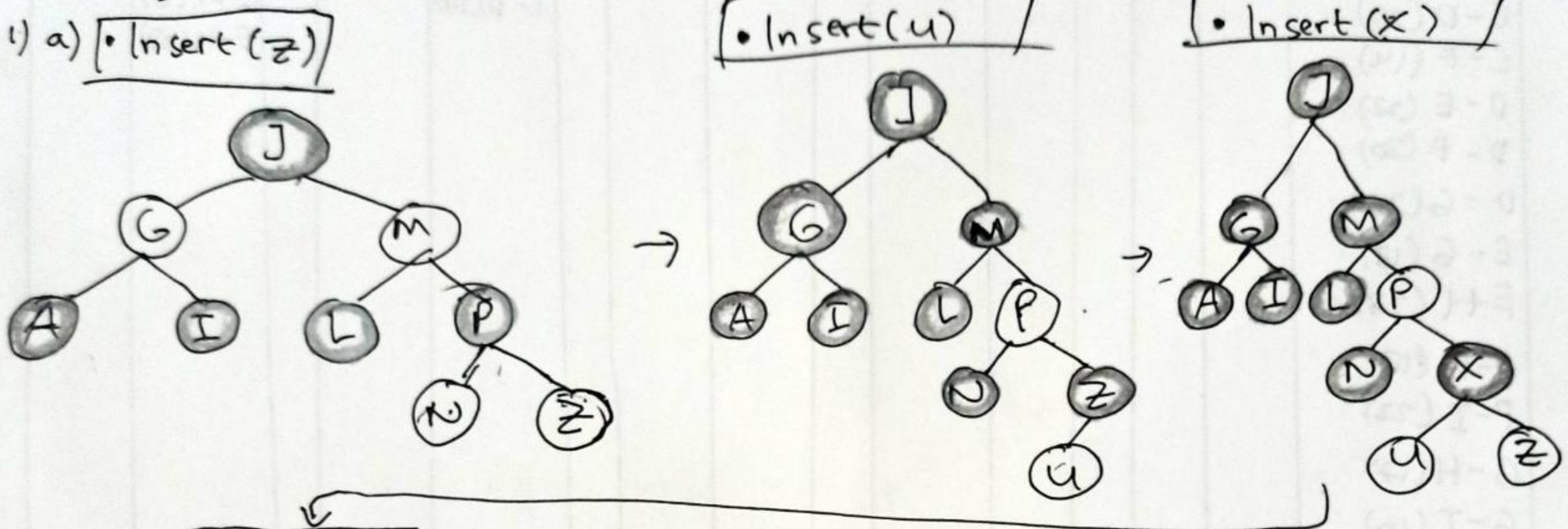


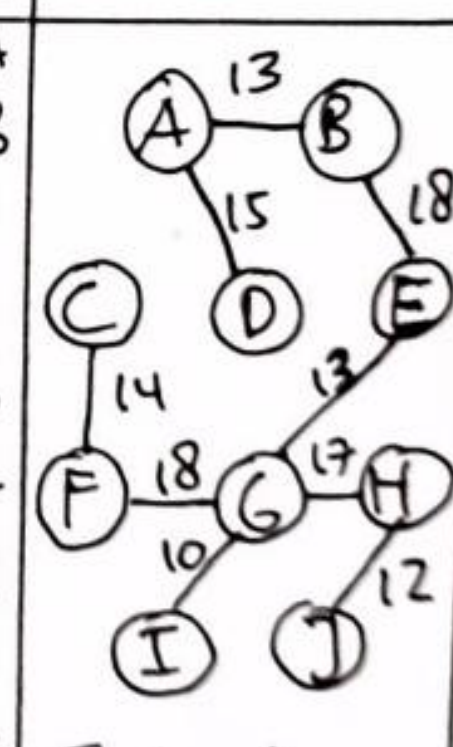
UAS Data Structure

Nama: Sendiawan
 NIM: 2301862733
 Kelas: LX01
 Dosen: Dion Darmawan
 Kode dosen: D4457

I) Essay
 1) a) Insert(z)



2)	Adjacency List	①			②			③			④		
		PQ	T	V	PQ	T	V	PQ	T	V	PQ	T	V
	A-B(13) A-C(22) A-D(15) B-D(26) B-E(18) C-D(28) C-F(14) D-E(22) D-F(20) D-G(26) E-G(13) E-H(28) F-G(18) F-I(22) G-H(17) G-I(10) G-J(21) H-J(12) I-J(26)	Null	Null		A-B(13) A-C(22) A-D(15)	Null	A	A-D(15) B-E(18) A-C(22) B-D(26)	A-B(13) B	A B D	B-E(13) D-F(20) A-C(22) D-E(22) B-D(26) D-G(26) C-D(28) E-H(28)	A-B(13) A-D(15) B-E(18)	A B D E

⑤			⑥			⑦			⑧			⑨			⑩			Result
PQ	T	V	PQ	T	V	PQ	T	V	PQ	T	V	PQ	T	V	PQ	T	V	
G-I(10) B-H(17) F-G(18) D-F(20) G-J(21) A-C(22) D-E(22) B-D(26) D-G(26) C-D(28) E-H(28)	A-B(13) A-D(15) B-E(18) E-G(13)	A B D E G	G-H(17) F-G(18) D-F(20) G-J(21) A-C(22) D-E(22) B-D(26) D-G(26) I-J(26) C-D(28) E-H(28)	A-B(13) A-D(15) B-E(18) E-G(13) G-I(10) D-E(22) B-D(26) D-G(26) I-J(26) C-D(28) E-H(28)	A B D E G I	H-J(12) A-B(13) A-D(15) B-E(18) G-J(21) E-G(13) A-C(22) G-I(10) D-E(22) G-H(17) F-I(22) B-D(26) D-G(26) I-J(26) C-D(28) E-H(28)	A B D E G I H		F-G(18) A-B(13) A-D(15) B-E(18) G-J(21) B-E(18) A-C(22) E-G(13) D-E(22) G-I(10) F-I(22) G-H(17) B-D(26) H-J(12) D-G(26) I-J(26) C-D(28) E-H(28)	A B D E G I H J		C-F(14) A-B(13) D-F(20) A-D(15) B-E(18) D A-C(22) E-G(13) D-E(22) G-I(10) F-I(22) G-H(17) B-D(26) H-J(12) D-G(26) F-G(18) I-J(26) C-D(28) E-H(28)	A B D E G I H J F		D-F(20) A-B(13) G-J(21) A-D(15) B-E(18) D-E(22) E-G(13) F-I(22) G-I(10) B-D(26) G-H(17) H-J(12) D-G(26) F-G(18) C-D(28) C-F(14) E-H(28)	A B D E G I H J F C		 <p>Total weight = 130</p>

3) ~~True~~

a) The smallest keys will always be stored in the left most leaf nodes of the tree (True)

- Karena rotasi - rotasi dari RBT dalam insert (maupun tdk terjadi rotasi) letak ~~ke~~ smallest keys pasti distore di node node leaf paling kiri tree

b) False (bisa semua hitam jd tdk 'always')

c) False, (seharusnya $= O(\log n)$)

d) False, seharusnya $O(\log N)$

4) ~~False~~

a) True, (AVL dan RBT ~~memiliki~~ terdapat rotasi untuk self balance)

b) True, ~~AVL dan RBT adalah Binary Tree~~ (karena memiliki paling byk 2 node)

~~c) True, karna kedua-duanya dpt implementasi binary tree~~

~~d) True, karna keduanya sama~~

e) False, krn B-Tree memiliki node bisa lebih dari 2 sehingga

bln binary Tree (heap adlh binary Tree)

d) True, tdk menerima informasi yang sama