

* Binary Search Tree

Insert the following values into it according to given order

1	11	43	19	17	2	3	4
---	----	----	----	----	---	---	---

(a) Insert 1
①

(b) Insert 11
① — ⑪

(c) Insert 43
① — ⑪ — ④③

(d) Insert 19
① — ⑪ — ④③ — ⑱

(e) Insert 17
① — ⑪ — ④③ — ⑱ — ⑰

(f) Insert 2
① — ⑪ — ④③ — ⑱ — ⑰ — ②

(g) Insert 3
① — ⑪ — ④③ — ⑱ — ⑰ — ② — ③

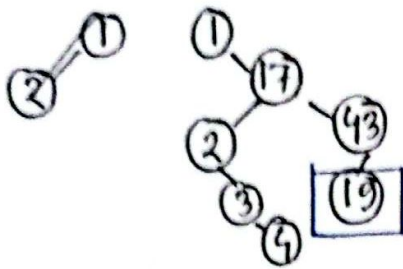
(h) Insert 4
① — ⑪ — ④③ — ⑱ — ⑰ — ② — ③ — ④

(i) Delete 11

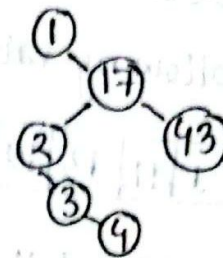
① — ⑪ — ④③ — ⑱ — ⑰ — ② — ③ — ④
→ left child & right child present — go to the rightmost min

① — ⑰ — ④③ — ⑱ — ② — ③ — ④

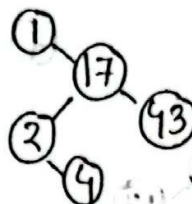
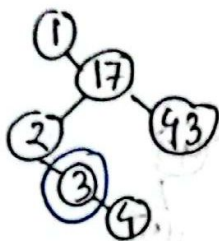
(b) Delete 19



no
single child
just delete



(c) Delete 3



Heap

* max heap tree. Insert the following values.

1, 2, 3, 4, 55, 6, 37, 12, 82, 32, 11, 43, 19, 17

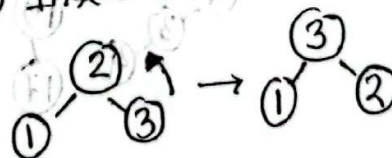
① (Insert) 1



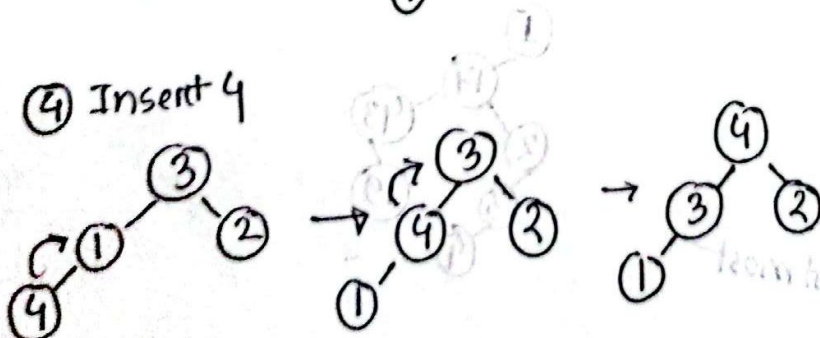
② Insert 2



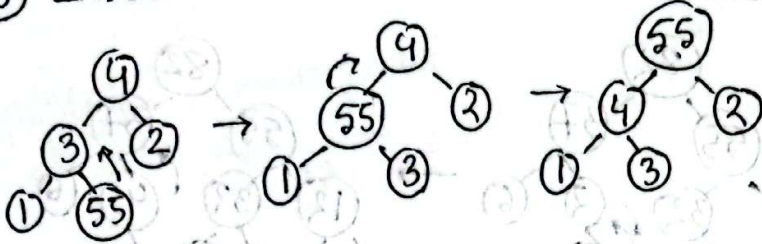
③ Insert 3



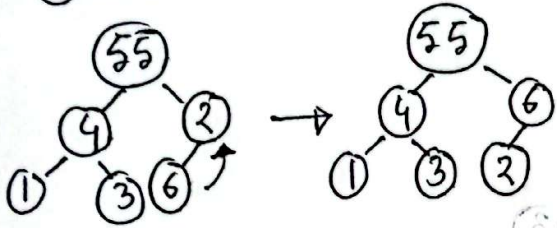
④ Insert 4



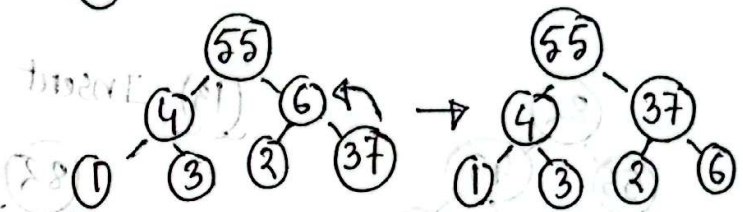
⑤ Insert 55



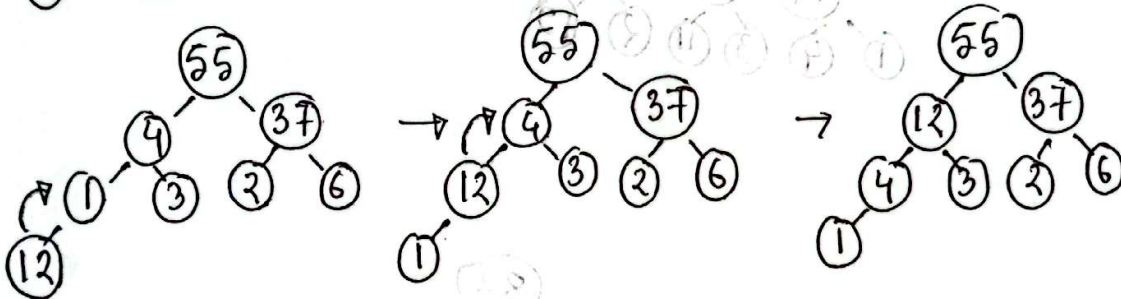
⑥ Insert 6



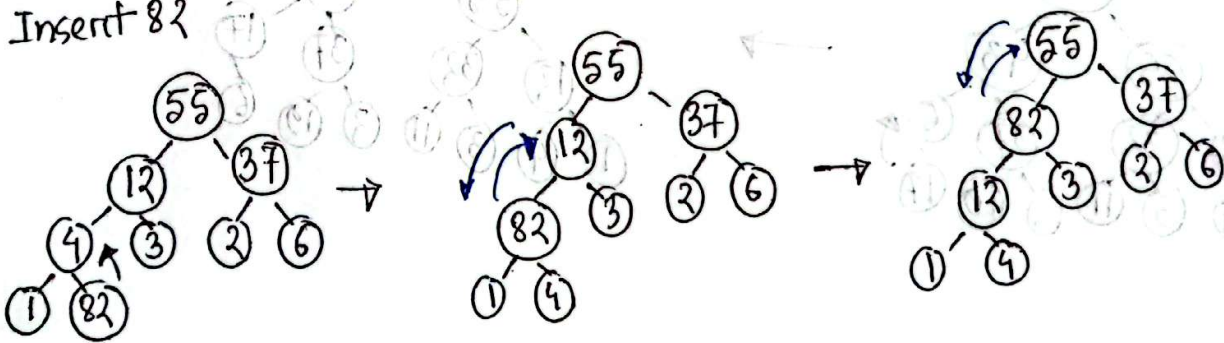
⑦ Insert 37



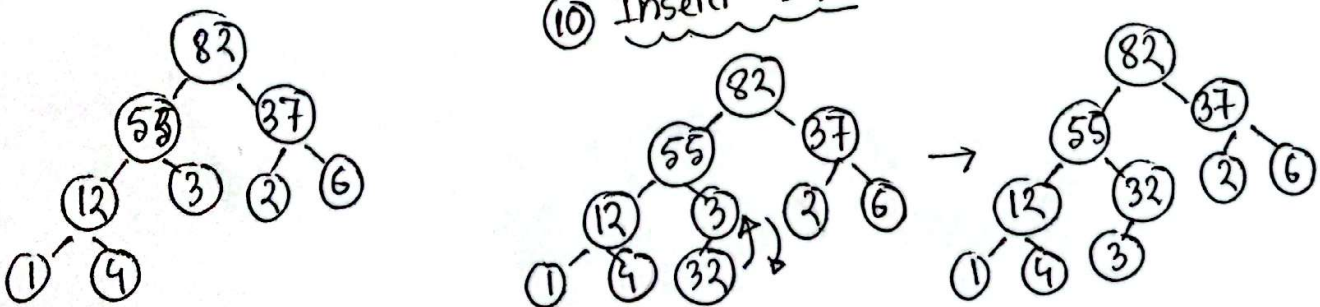
⑧ Insert 12



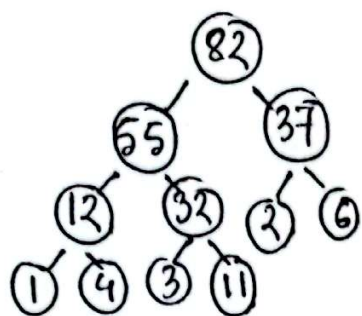
⑨ Insert 82



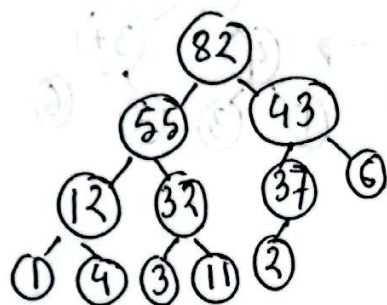
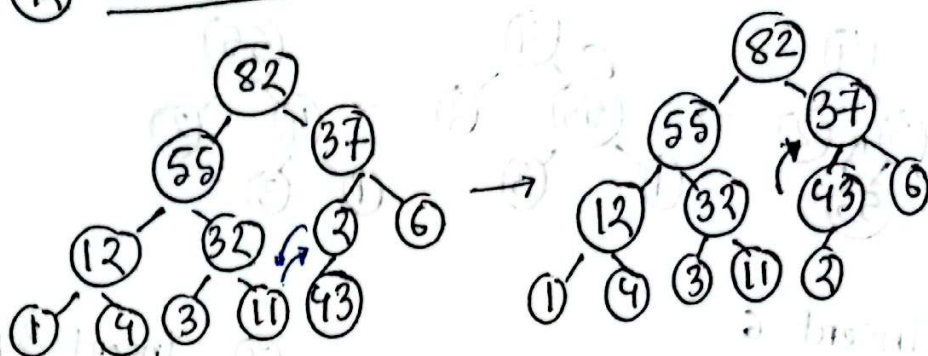
⑩ Insert 32



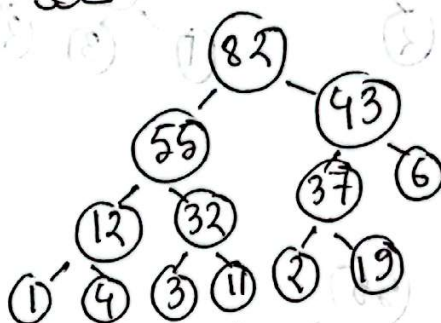
(11) Insert 11



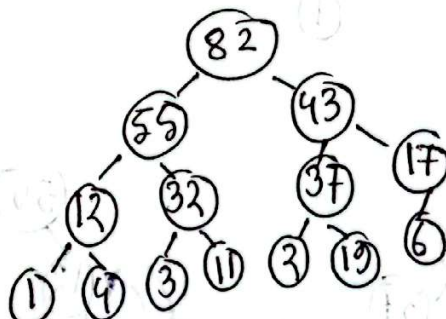
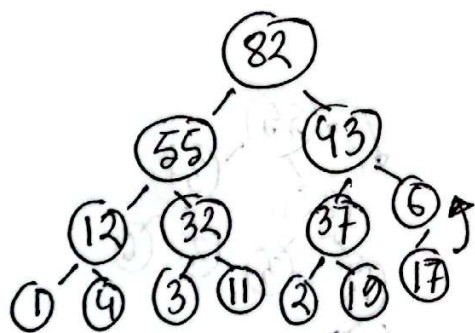
(12) Insert 43



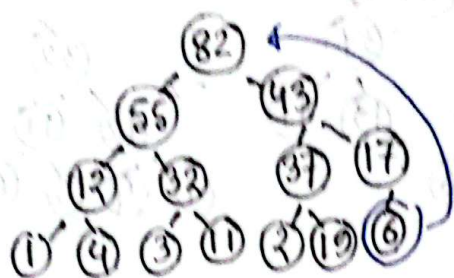
(13) Insert 19



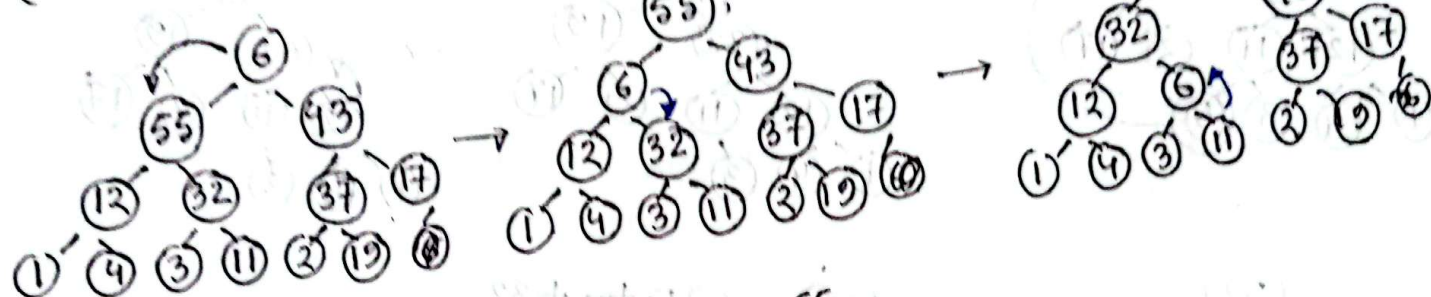
(14) Insert 17



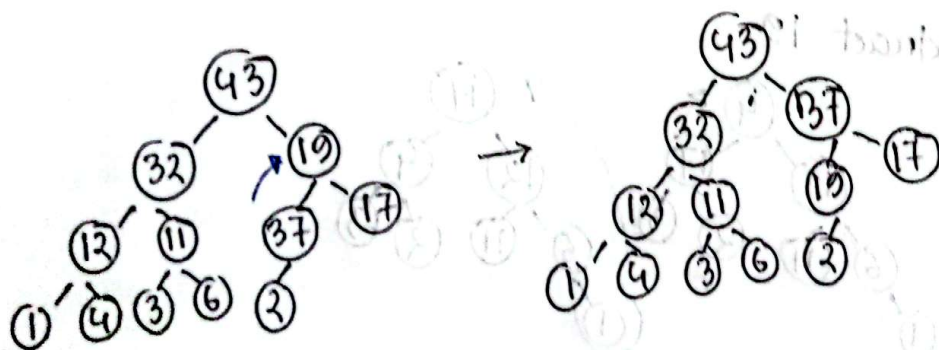
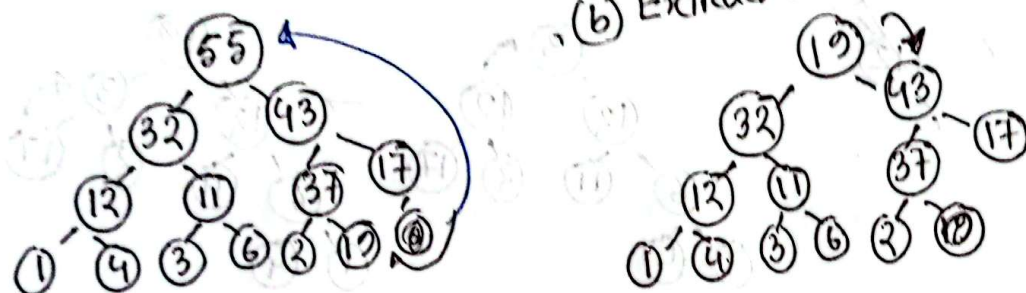
8) Extract the maxheap until it becomes empty.



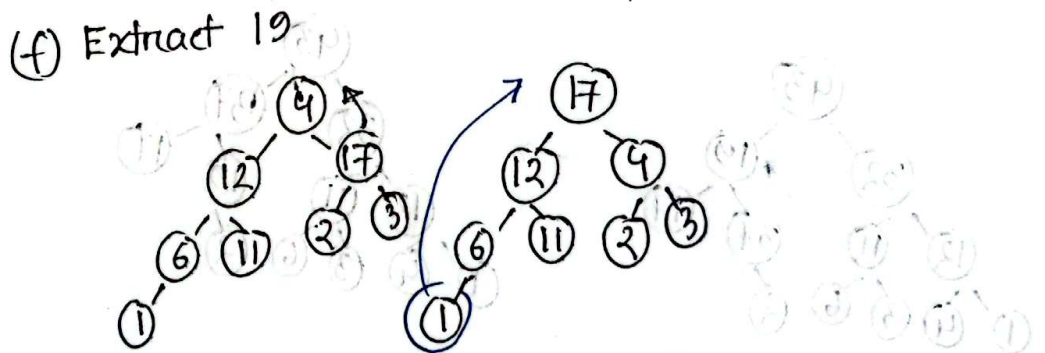
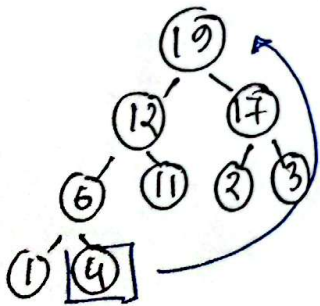
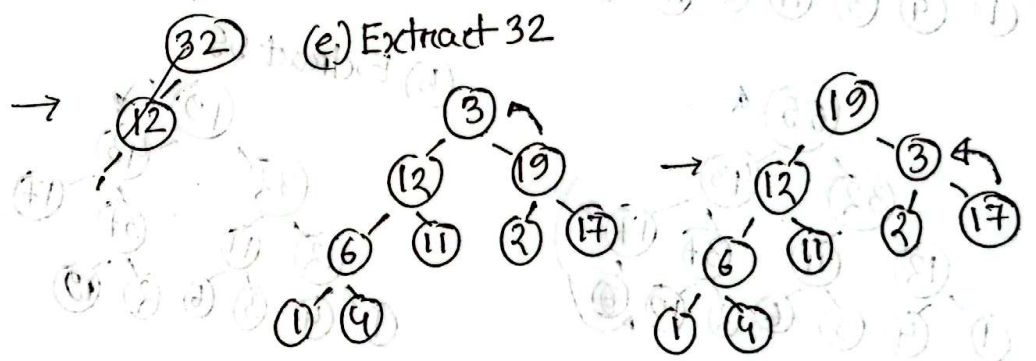
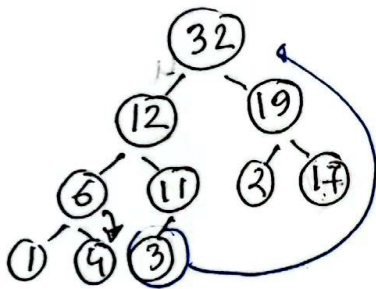
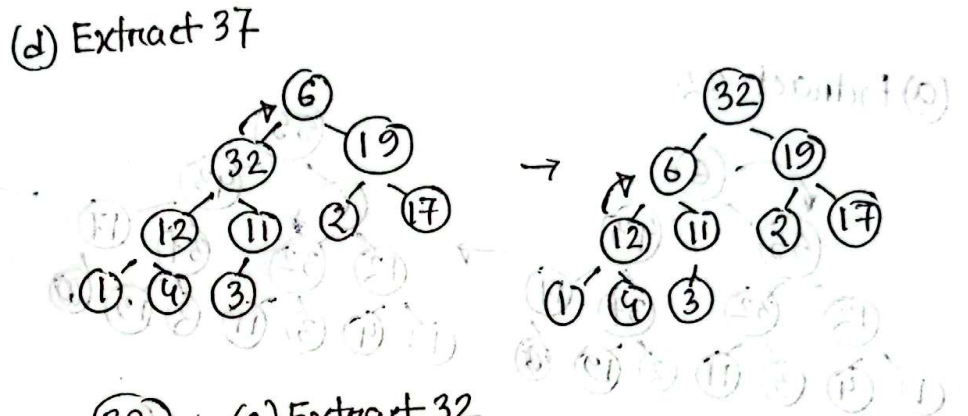
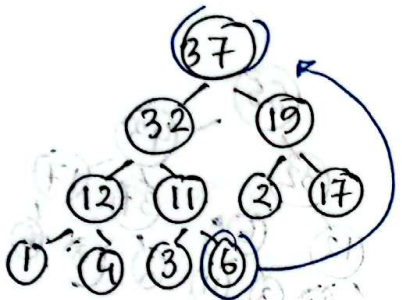
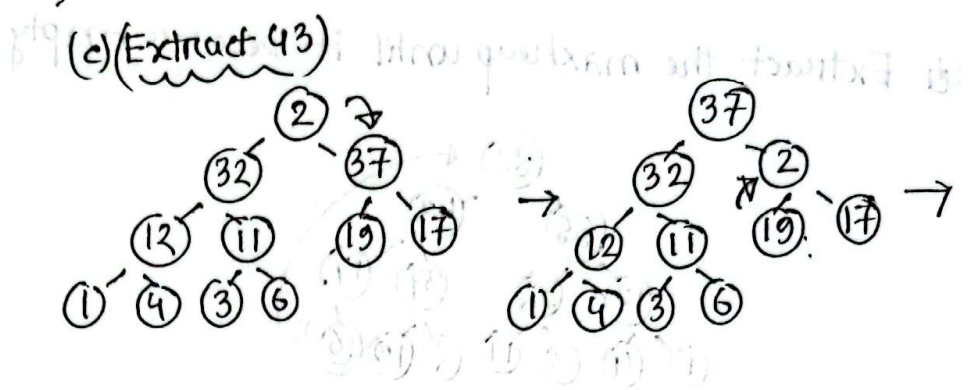
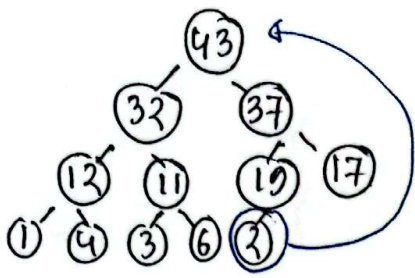
(a) Extract 82.



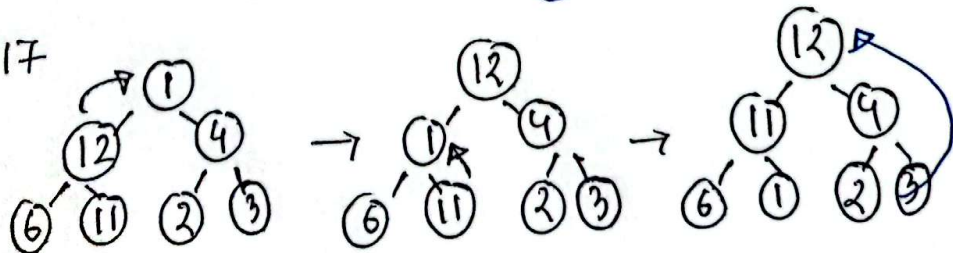
(b) Extract 55



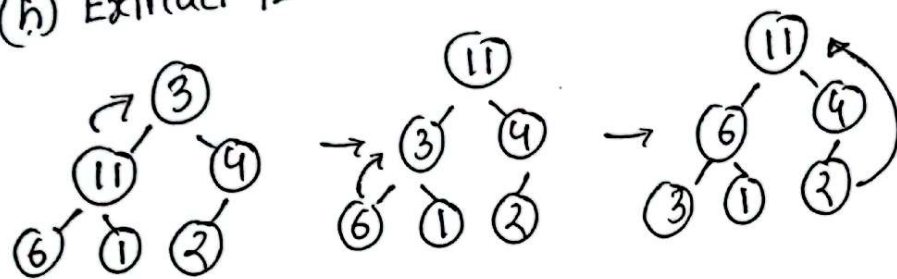
c



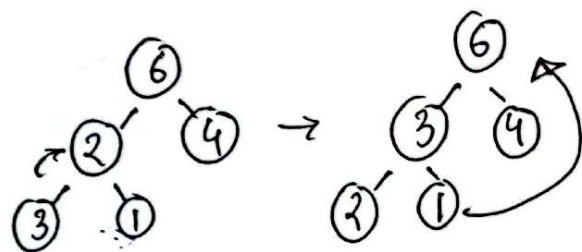
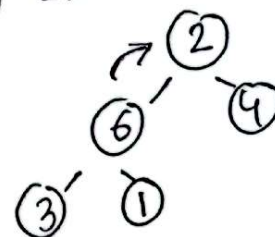
(g) Extract 17



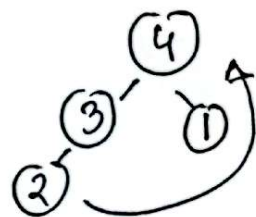
(h) Extract 12



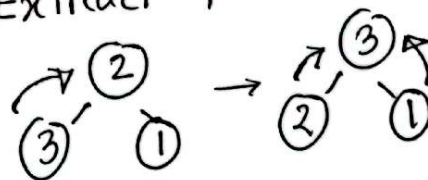
(i) Extract 11



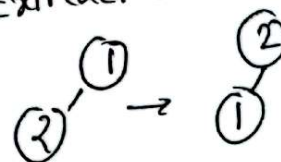
(j) Extract 6



(k) Extract 4



(l) Extract 3



(m) Extract 2

①

(n) Extract

①