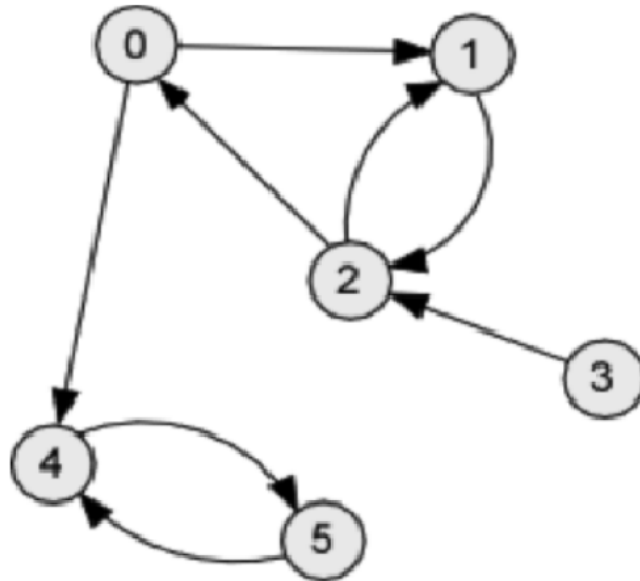


1 Graph Representations



In the parts below, enter the adjacency list corresponding to each node.

For every part, your answer should be **sorted** by least to greatest with a **single space** between each item. For example, a node A that has edges to nodes D, C, and B would be input as: B C D

- What is the adjacency list for node 0?
- What is the adjacency list for node 1?
- What is the adjacency list for node 2?
- What is the adjacency list for node 4?

2 Heaps

We have the following heap, representing a Min PQ: [-, 1, 4, 6, 7, 10, 12, 15, 16]. Here, - represents null.

a) What is the left child of element 4?

- 6 7 10 12

b) What is the right child of element 6?

- 7 10 12 15

c) Suppose we insert 0. What is the resulting state of the heap?

- [-, 1, 0, 6, 4, 10, 12, 15, 16, 7]

- [-, 0, 1, 6, 4, 10, 12, 15, 16, 7]

- [-, 0, 1, 4, 6, 10, 12, 15, 16, 7]

- [-, 0, 1, 4, 6, 7, 10, 12, 15, 16]

- None of the above

d) Consider the initial state of the heap again. [-, 1, 4, 6, 7, 10, 12, 15, 16]

Suppose we call `removeMin()`. What is the resulting state of the heap?

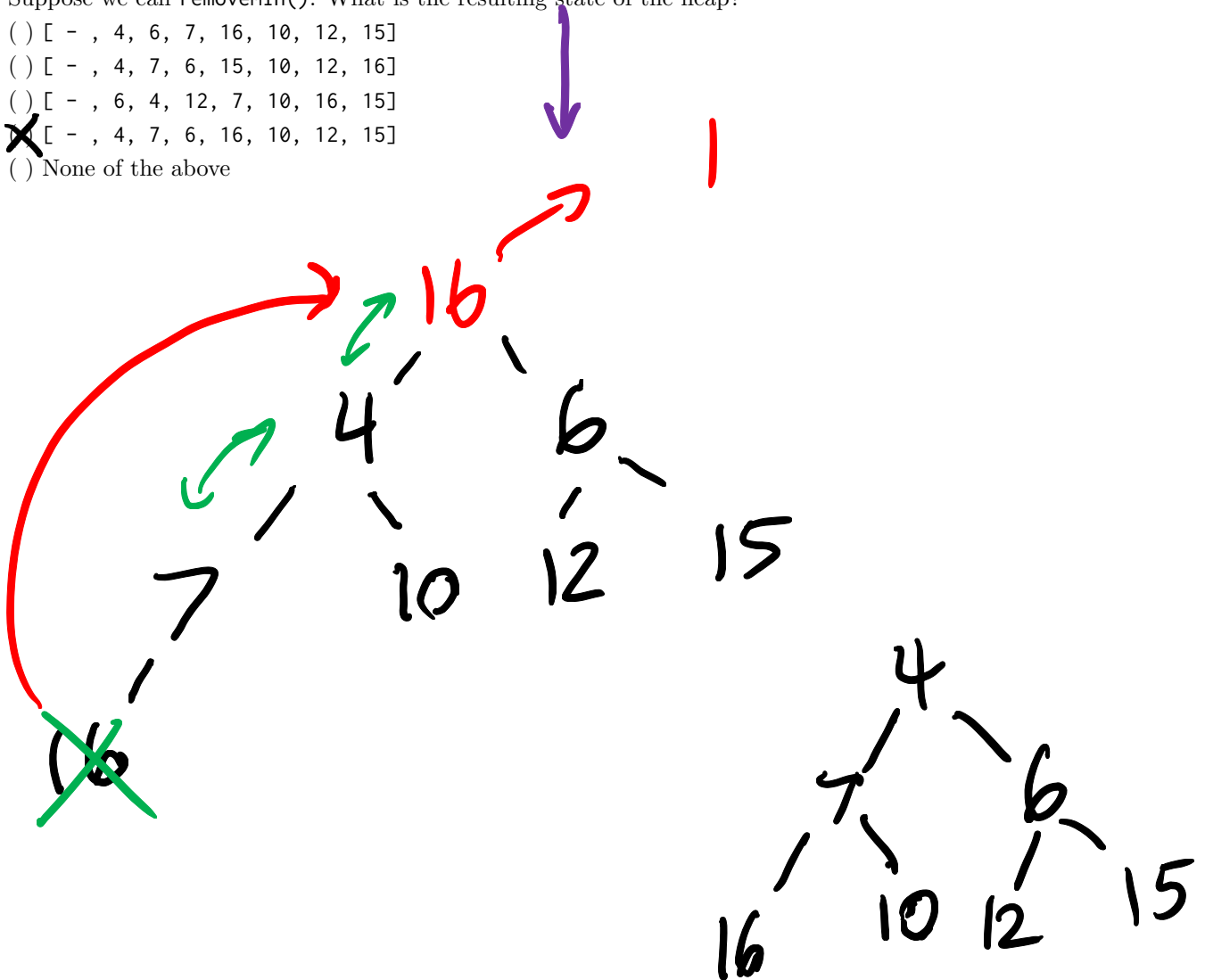
- [-, 4, 6, 7, 16, 10, 12, 15]

- [-, 4, 7, 6, 15, 10, 12, 16]

- [-, 6, 4, 12, 7, 10, 16, 15]

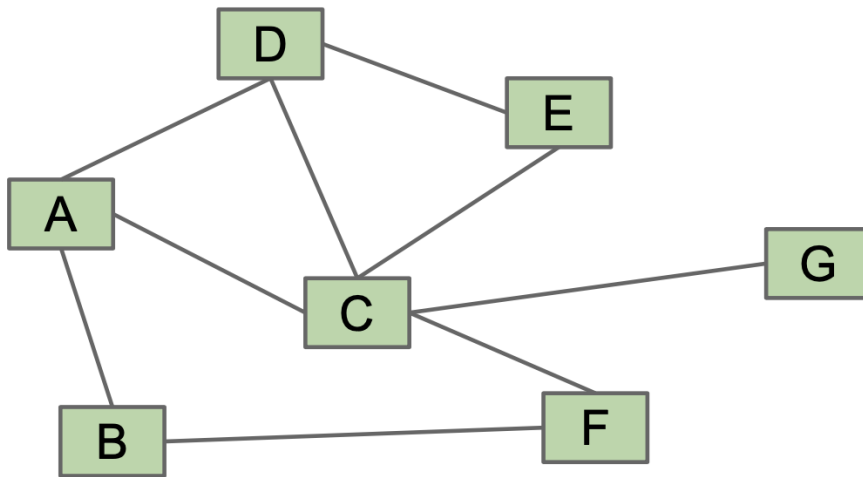
- [-, 4, 7, 6, 16, 10, 12, 15]

- None of the above



3 Graphs

Suppose we have the graph below.



For all parts below, assume we break ties alphabetically.

- a) What is the order that vertices are visited if we run DFS preorder starting on vertex A? Enter your answer as a space separated list, e.g. A B C D E F G.
- b) What is the order that vertices are visited if we run DFS postorder starting on vertex A? Enter your answer as a space separated list, e.g. A B C D E F G.
- c) What is the order that vertices are visited if we run BFS starting on vertex A? Enter your answer as a space separated list, e.g. A B C D E F G.

2 Heap Mystery

We are given the following array representing a min-heap where each letter represents a **unique** number. Assume the root of the min-heap is at index zero, i.e. A is the root. Note that there is **no** significance of the alphabetical ordering, i.e. just because B precedes C in the alphabet, we do not know if B is less than or greater than C.

Array: [A, B, C, D, E, F, G]

Four unknown operations are then executed on the min-heap. An operation is either a `removeMin` or an `insert`. The resulting state of the min-heap is shown below.

Array: [A, E, B, D, X, F, G]

(a) Determine the operations executed and their appropriate order. The first operation has already been filled in for you!

1. `removeMin()`
2. insert (X)
3. remove (C)
4. insert (A)

(b) Fill in the following comparisons with either $>$, $<$, or $?$ if unknown. Note that this question does not assume a specific ordering of operations from the previous part, i.e. we don't know which of the two possible

1. X ? D
2. X > C
3. B > C
4. G < X

3. remove C
4. insert A
2. insert X

Major Differences:

X is added but somehow goes to the left - ~~too~~
 A is readded and stays on top
 C is gone (somewhere it needs to be at the top)

