

AMCAT Heaps Questions

Question 1

Which data structure is needed to convert infix notations to postfix notations

- (A) linear list
- (B) tree
- (C) stack
- (D) queue

Answer: Option C

Question 2

Which will be the best to implement the following a car comes at a petrol station and waits. The next car to get its Gas filled should be the one which has waited the longest time and thus is given priority?

- (A) Binary Trees
- (B) Heaps
- (C) m-way Trees
- (D) Binary Search Tree

Answer: Option B

Question 3

Which sorting method is slowest?

- (A) Quick sort
- (B) Heap sort
- (C) Shell sort
- (D) Bubble sort

Answer: Option D

Question 4

For a hashing table what is the time complexity?

- (A) $O(1)$
- (B) $O(n^2)$
- (C) $O(\log n)$
- (D) $O(n)$

Answer: Option A

Question 5

Every element of a data structure has an address and a key associated with it. A search mechanism deals with two or more values assigned to the same address by using the key. What

is this search mechanism?

- (A) Linear Search
- (B) Binary search
- (C) Hash Coded Search
- (D) None of these

Answer: Option C

Question 6

What is the time complexity of Build Heap operation. Build Heap is used to build a max(or min) binary heap from a given array. Build Heap is used in Heap Sort as a first step for sorting.

- (A) $O(n \log n)$
- (B) $O(n^2)$
- (C) $O(\log n)$
- (D) $O(n)$

Answer: Option D

Explanation:

Following is algorithm for building a Heap of an input array A.

```
BUILD-HEAP(A)
  heapsize := size(A);
  for i := floor(heapsize/2) downto 1
    do HEAPIFY(A, i);
  end for
END
```

Although the worst case complexity looks like $O(n \log n)$, upper bound of time complexity is $O(n)$

Question 7

In a binary max heap containing n numbers, the smallest element can be found in time?

- (A) $O(n)$
- (B) $O(\log n)$
- (C) $O(\log \log n)$
- (D) $O(1)$

Answer: Option A

Question 8

Suppose we are sorting an array of eight integers using heapsort, and we have just finished some heapify (either maxheapify or minheapify) operations. The array now looks like this: 16 14 15 10 12 27 28 How many heapify operations have been performed on root of heap?

- (A) 1
- (B) 2

(C) 3 or 4

(D) 5 or 6

Answer: Option B

Explanation:

In Heap sort, we first build a heap, then we do following operations till the heap size becomes 1. a) Swap the root with last element b) Call heapify for root c) reduce the heap size by 1. In this question, it is given that heapify has been called few times and we see that last two elements in given array are the 2 maximum elements in array. So situation is clear, it is maxheapify which has been called 2 times.

Question 9

Consider a binary max-heap implemented using an array. Which one of the following array represents a binary max-heap?

(A) 25,12,16,13,10,8,14

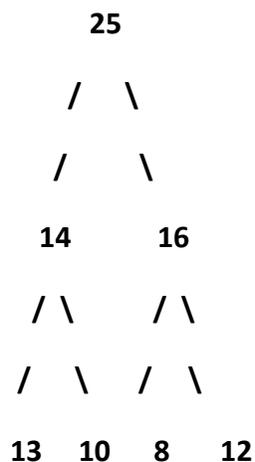
(B) 25,12,16,13,10,8,14

(C) 25,14,16,13,10,8,12

(D) 25,14,12,13,10,8,16

Answer: Option C

Explanation: A tree is max-heap if data at every node in the tree is greater than or equal to its children's data. In array representation of heap tree, a node at index i has its left child at index $2i + 1$ and right child at index $2i + 2$.



Question 10

What is the content of the array after two delete operations on the correct answer to the previous question?

(A) 14,13,12,10,8

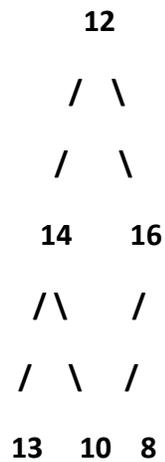
(B) 14,12,13,8,10

(C) 14,13,8,12,10

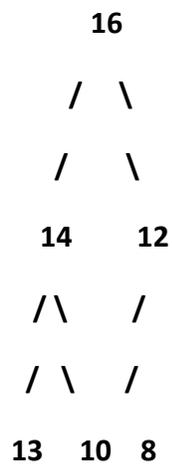
(D) 14,13,12,8,10

Answer: Option D

Explanation: For Heap trees, deletion of a node includes following two operations. 1) Replace the root with last element on the last level. 2) Starting from root, heapify the complete tree from top to bottom.. Let us delete the two nodes one by one: 1) Deletion of 25: Replace 25 with 12



Since heap property is violated for root (16 is greater than 12), make 16 as root of the tree.



2) Deletion of 16: Replace 16 with 8



/ \

/ \

13 10

Heapify from root to bottom.

14

/ \

/ \

8 12

/ \

/ \

13 10

14

/ \

/ \

13 12

/ \

/ \

8 10