CSC263 Tutorial #2

Heaps

January 20, 2023

Things that will be covered in this tutorial

Things that will be covered in this tutorial

- ⋆ What's a priority queue?
- ⋆ What's a (max-)heap? How can I access elements of heaps?
- ⋆ What's a "ternary max-heap"?

Priority Queues

Question: What's a priority queue? What operations are supported by priority queues?

Priority Queues

Question: What's a priority queue? What operations are supported by priority queues?
Answer: A priority queue is a collection of elements, with a priority associated with each element.

Priority Queues

Question: What's a priority queue? What operations are supported by priority queues?
Answer: A priority queue is a collection of elements, with a priority associated with each element. The following operations are supported:

- `Insert(Q, x, k)`: Inserts `x` into the priority queue `Q` with priority `k`.

Priority Queues

Question: What's a priority queue? What operations are supported by priority queues?
Answer: A priority queue is a collection of elements, with a priority associated with each element. The following operations are supported:

- `Insert(Q, x, k)`: Inserts `x` into the priority queue `Q` with priority `k`.
- `Max(Q)`: Return an element with maximum priority in `Q`.[1]

[1] In the case of a tie, return an arbitrary element with maximum priority.

Priority Queues

Question: What's a priority queue? What operations are supported by priority queues?
Answer: A priority queue is a collection of elements, with a priority associated with each element. The following operations are supported:

- ⋆ `Insert(Q, x, k)`: Inserts `x` into the priority queue `Q` with priority `k`.
- ⋆ `Max(Q)`: Return an element with maximum priority in `Q`.[1]
- ⋆ `ExtractMax(Q)`: Extract an element with maximum priority from `Q`.

[1] In the case of a tie, return an arbitrary element with maximum priority.

Priority Queues

Question: What's a priority queue? What operations are supported by priority queues?
Answer: A priority queue is a collection of elements, with a priority associated with each element. The following operations are supported:

- ⋆ `Insert(Q, x, k)`: Inserts `x` into the priority queue `Q` with priority `k`.
- ⋆ `Max(Q)`: Return an element with maximum priority in `Q`.[1]
- ⋆ `ExtractMax(Q)`: Extract an element with maximum priority from `Q`.
- ⋆ `IncreasePriority(Q, x, k)`: Incerase `x`'s priority to `k` in `Q`.

[1] In the case of a tie, return an arbirtary element with maximum priority.

Priority Queues

Question: What's a priority queue? What operations are supported by priority queues?
Answer: A priority queue is a collection of elements, with a priority associated with each element. The following operations are supported:

- ⋆ `Insert(Q, x, k)`: Inserts `x` into the priority queue `Q` with priority `k`.
- ⋆ `Max(Q)`: Return an element with maximum priority in `Q`.[1]
- ⋆ `ExtractMax(Q)`: Extract an element with maximum priority from `Q`.
- ⋆ `IncreasePriority(Q, x, k)`: Incerase `x`'s priority to `k` in `Q`.

[1] In the case of a tie, return an arbirtary element with maximum priority.

Priority Queues

Question: What's a priority queue? What operations are supported by priority queues?
Answer: A priority queue is a collection of elements, with a priority associated with each element. The following operations are supported:

- ⋆ `Insert(Q, x, k)`: Inserts `x` into the priority queue `Q` with priority `k`.
- ⋆ `Max(Q)`: Return an element with maximum priority in `Q`.[1]
- ⋆ `ExtractMax(Q)`: Extract an element with maximum priority from `Q`.
- ⋆ `IncreasePriority(Q, x, k)`: Incerase `x`'s priority to `k` in `Q`.

Question: What can a priority queue be used for?

[1] In the case of a tie, return an arbirtary element with maximum priority.

Priority Queues

Question: What's a priority queue? What operations are supported by priority queues?
Answer: A priority queue is a collection of elements, with a priority associated with each element. The following operations are supported:

- ⋆ `Insert(Q, x, k)`: Inserts `x` into the priority queue `Q` with priority `k`.
- ⋆ `Max(Q)`: Return an element with maximum priority in `Q`.[1]
- ⋆ `ExtractMax(Q)`: Extract an element with maximum priority from `Q`.
- ⋆ `IncreasePriority(Q, x, k)`: Incerase `x`'s priority to `k` in `Q`.

Question: What can a priority queue be used for?
Answer: Things like hospital triage, or your workload.

[1] In the case of a tie, return an arbirtary element with maximum priority.

Heaps

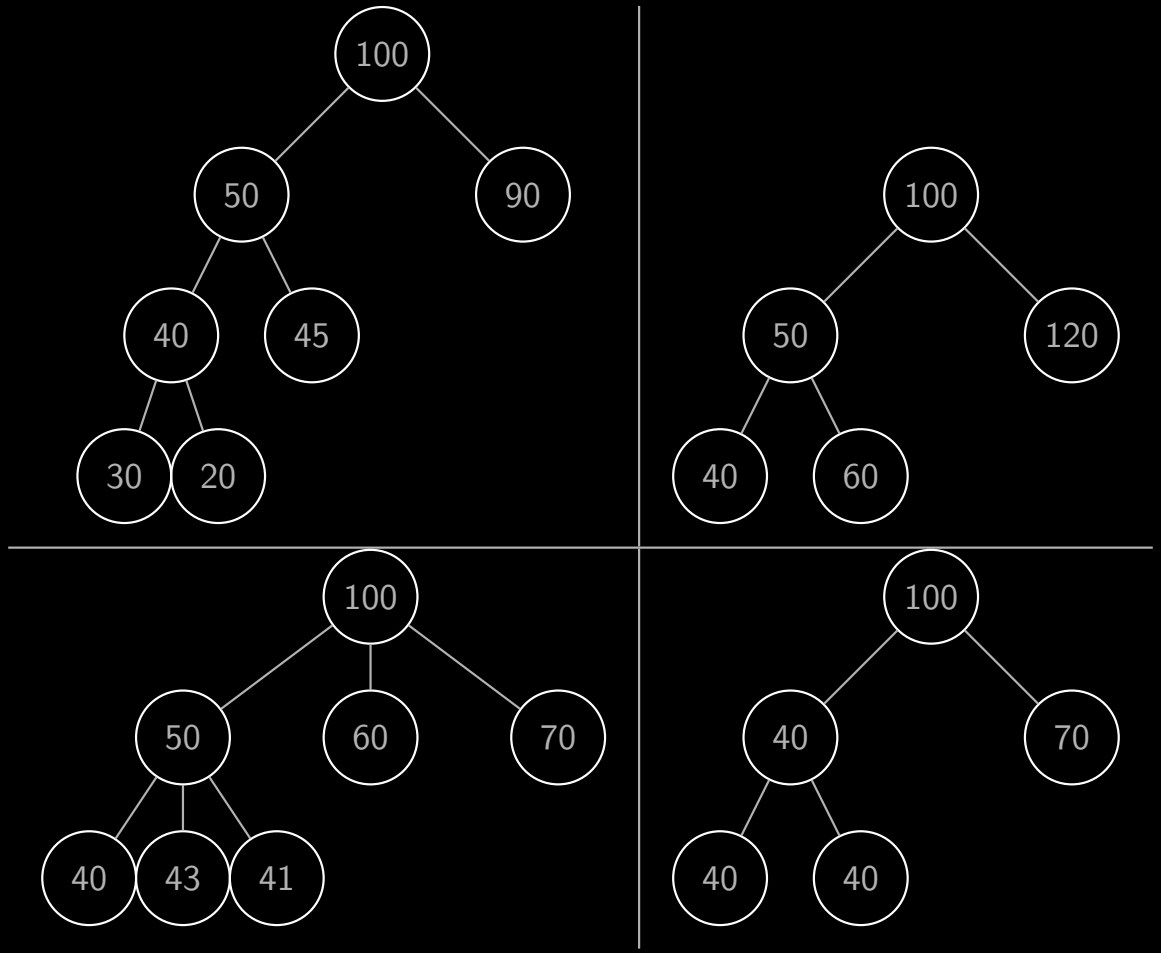
Heaps

Heaps implement the priority queue ADT.

Heaps

Heaps implement the priority queue ADT.

Question: Which of the following is a binary max-heap?



Tutorial Question 1

Tutorial Question: Alice claims that the minimum element of a binary max-heap must be one of its leaf nodes. Do you agree? Prove or disprove it.

Tutorial Question 1

Tutorial Question: Alice claims that the minimum element of a binary max-heap must be one of its leaf nodes. Do you agree? Prove or disprove it.

Hint: There are two correct answers, depending on the assumptions you make.

Tutorial Question 2

Tutorial Question: Bob claims that the median element of a binary max-heap must be one of its leaf nodes. Do you agree? Prove or disprove it.

Tutorial Question 2

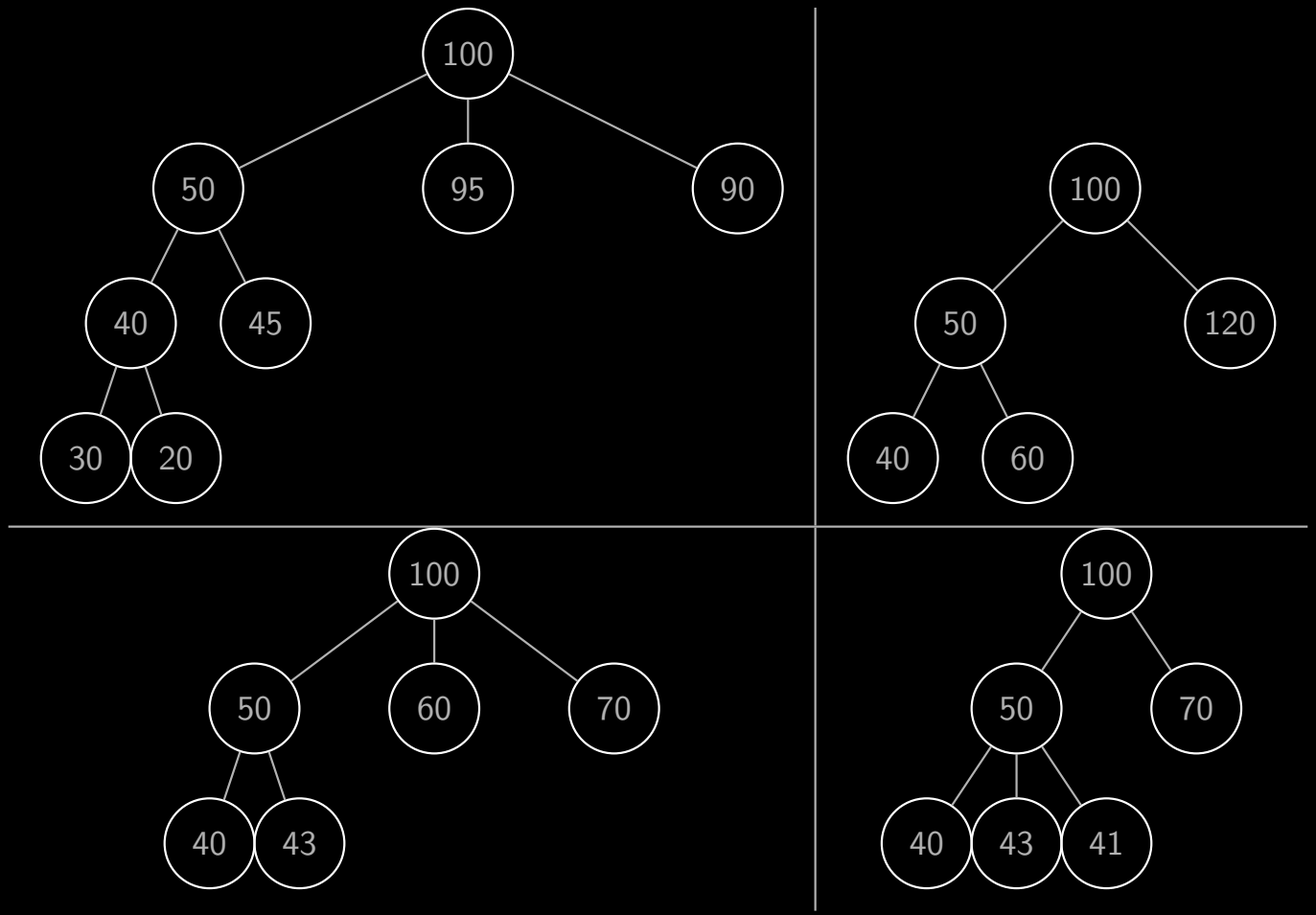
Tutorial Question: Bob claims that the median element of a binary max-heap must be one of its leaf nodes. Do you agree? Prove or disprove it.

Hint: There is only one correct answer this time.

Ternary Heaps

Ternary Heaps

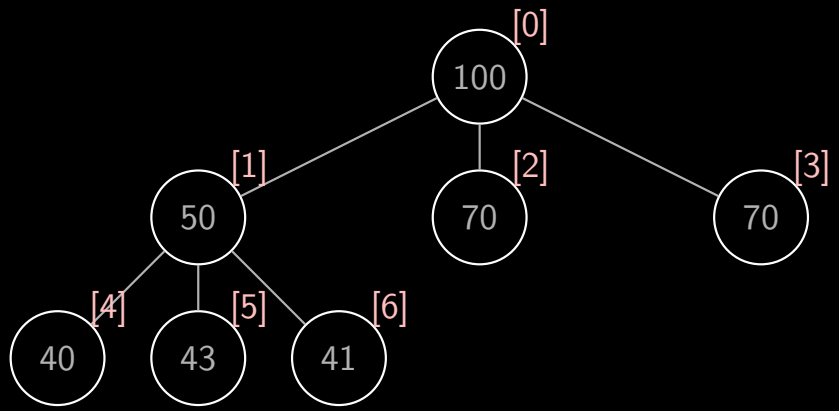
Like binary heaps, but with three children (instead of two)!
Question: Which of the following is a ternary max-heap?



Ternary Heaps

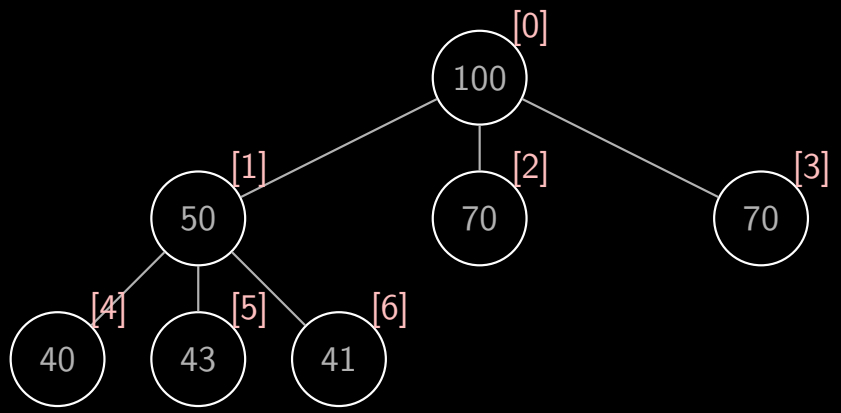
Ternary Heaps

Ternary Heaps are represented with arrays.



Ternary Heaps

Ternary Heaps are represented with arrays.



Question:

- ⋆ Which index is the left child of [0]?
- ⋆ Which index is the middle child of [1]?
- ⋆ Which index is the parent of [2]?
- ⋆ Which index is the parent of [6]?

Ternary Heaps

Tutoiral Question: How to check when an array is a ternary heap?

Ternary Heaps

Tutoiral Question: How to check when an array is a ternary heap?

```
ArrayIsBinaryMaxHeap(A):
    n = len(A)
    for i in 0..(A//2-1):
        if A[i] < A[2i] or A[i] < A[2i+1]:
            return False
    return True
```

Ternary Heaps

Tutoiral Question: How to extract max from a ternary heap?

Ternary Heaps

Tutoiral Question: How to extract max from a ternary heap?

```
ExtractMax(A):
    n = len(A)
    swap A[0] with A[n]
    max = A.pop()
    BubbleDown(A[0])
    return max
```

Ternary Heaps

Tutoiral Question: How to insert into a ternary heap?

Ternary Heaps

Tutoiral Question: How to insert into a ternary heap?

```
Insert(A, x, k):
    x.priority = k
    A.insert(x)
    BubbleUp(x)
```