FF Hashing & Heaps

OCT 20 2023

Create a video explaining and implementing the code for the following instructions

P1a: Emergency Room Organization

You are hired to automate the order in which patients at the local ER are treated. Design a program, utilizing a heap, in which you arrange the patient's order of treatment. Output the order in which they are treated.

Speed is the priority with this task. Patients must be inserted into their correct position in line with O(logn) time.

Injuries are denoted on a scale of 1 - 20 with 20 being critical condition and 1 being unsevere. Severe injuries must be treated first.

P1a: Emergency Room Organization

Input

```
Usman : 12
Francis : 11
TNP
Colby : 13
Leon : 4
TNP
TNP
TNP
Adesanya : 20
TNP
```

Output

```
Usman is treated
Colby is treated
Francis is treated
Leon is treated
Adesanya is treated
```

TNP = “Treat Next Patient”

Print the rest of the patients, in order, when the end of the input is reached.

P1a: Emergency Room Organization

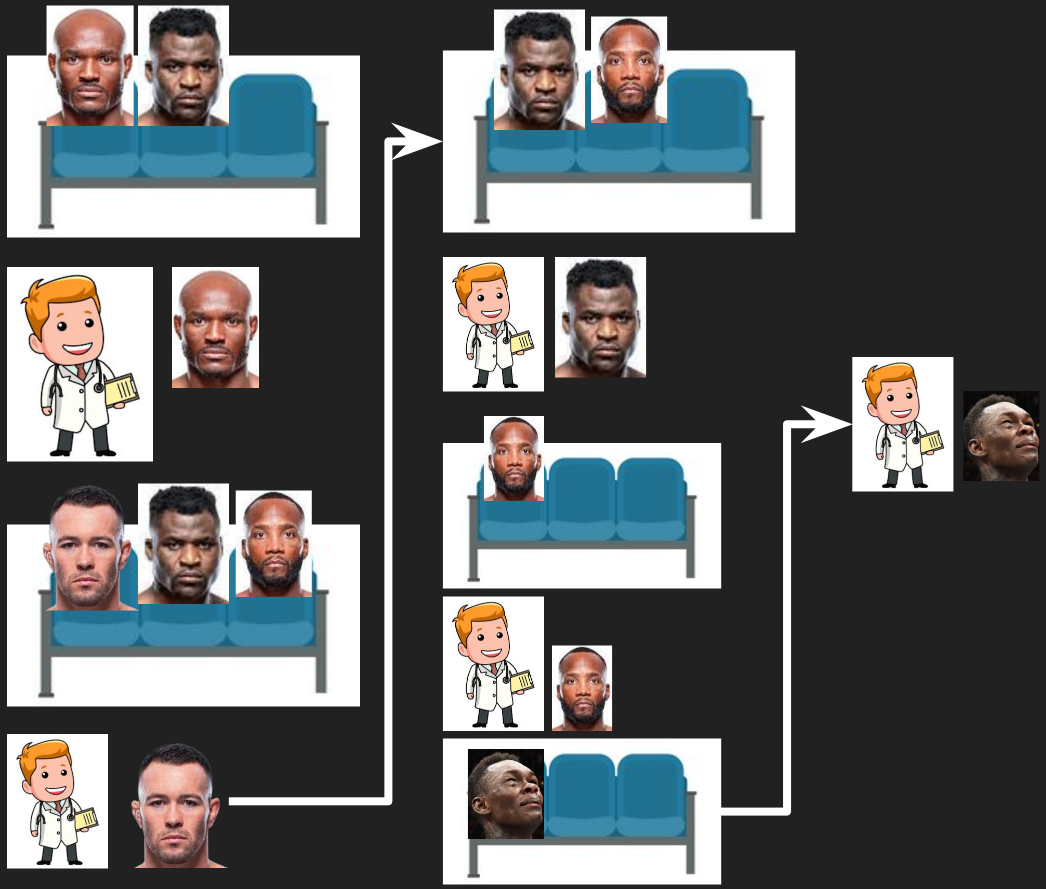
Input

Usman : 12
Francis : 11
TNP
Colby : 13
Leon : 4
TNP
TNP
TNP
Adesanya : 20
TNP

Output

Usman is treated
Colby is treated
Francis is treated
Leon is treated
Adesanya is treated

Note: →
Order of patients in waiting room (blue chairs) are organized in descending order and is not representative of what is expected in the solution.



P1b: Emergency Room Organization

If done correctly, you will have a program that utilizes a heap in some way to access and treat the most severe injury in the ER at the time.

This method however, is illegible to those who can’t read heaps. Draw and explain how you can transcribe some heap into a more legible sorted array. Explain the function(s), and how it works, used in this algorithm.

You must show every iteration of this algorithm in your video.

You may use coding snippets found online to help explain the process. You must use your own visual aid when demonstrating the algorithm. Visual aid can be drawings/ppts/etc.

P2: Hiking Trail Shortcuts

You are charting a map of a hiking trail as you traverse through it. The trail visits one or more landmarks, denoted by n linked list nodes. Each of these landmarks have two possible directions a hiker may take: A path pointing to the next landmark and a shortcut to some landmark in the hiking trail.

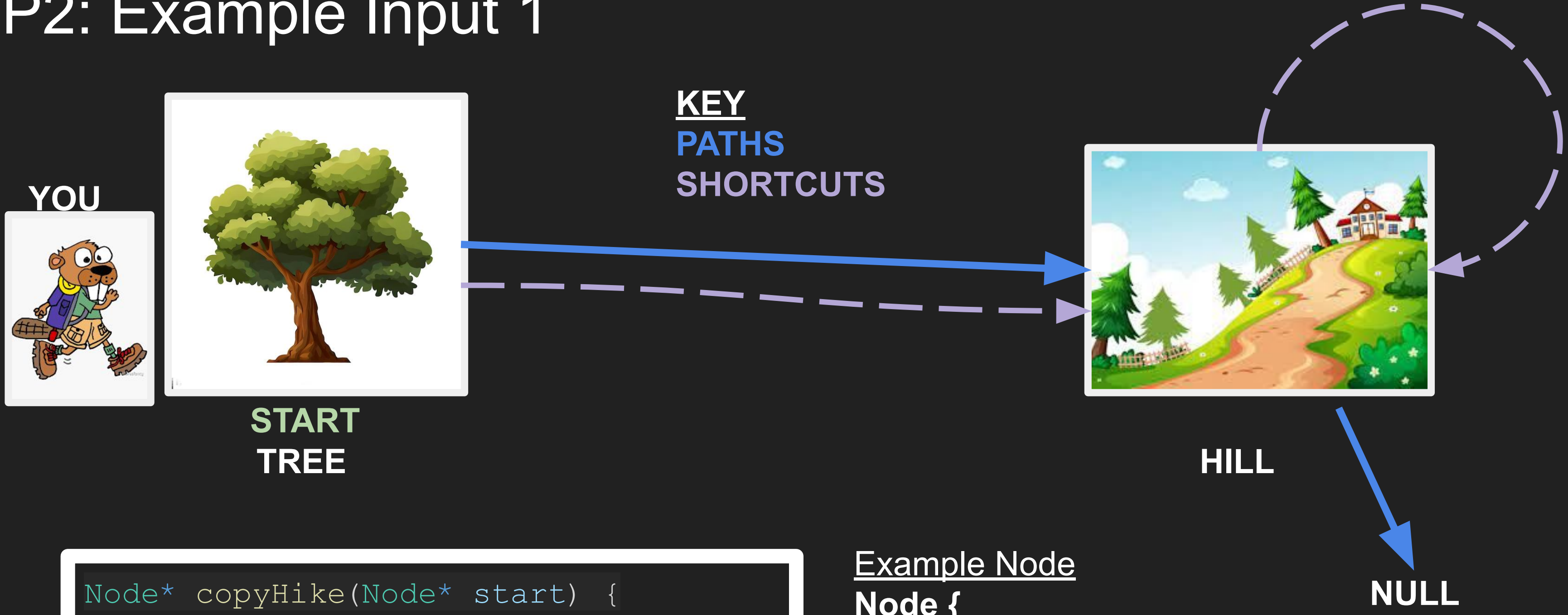
Design a program that creates a **deep copy** the trail, including both the shortcuts and paths, in O(N) time.

Not every landmark has to contain a shortcut, but every node must have a path to the next landmark.

None of the paths & shortcuts in the copied hiking trail should point to any landmarks from the original trail.

Your function will **only** have access to the starting landmark.
Return the start node of your copied trail.

P2: Example Input 1



YOU

START
TREE

KEY
PATHS
SHORTCUTS

HILL

NULL

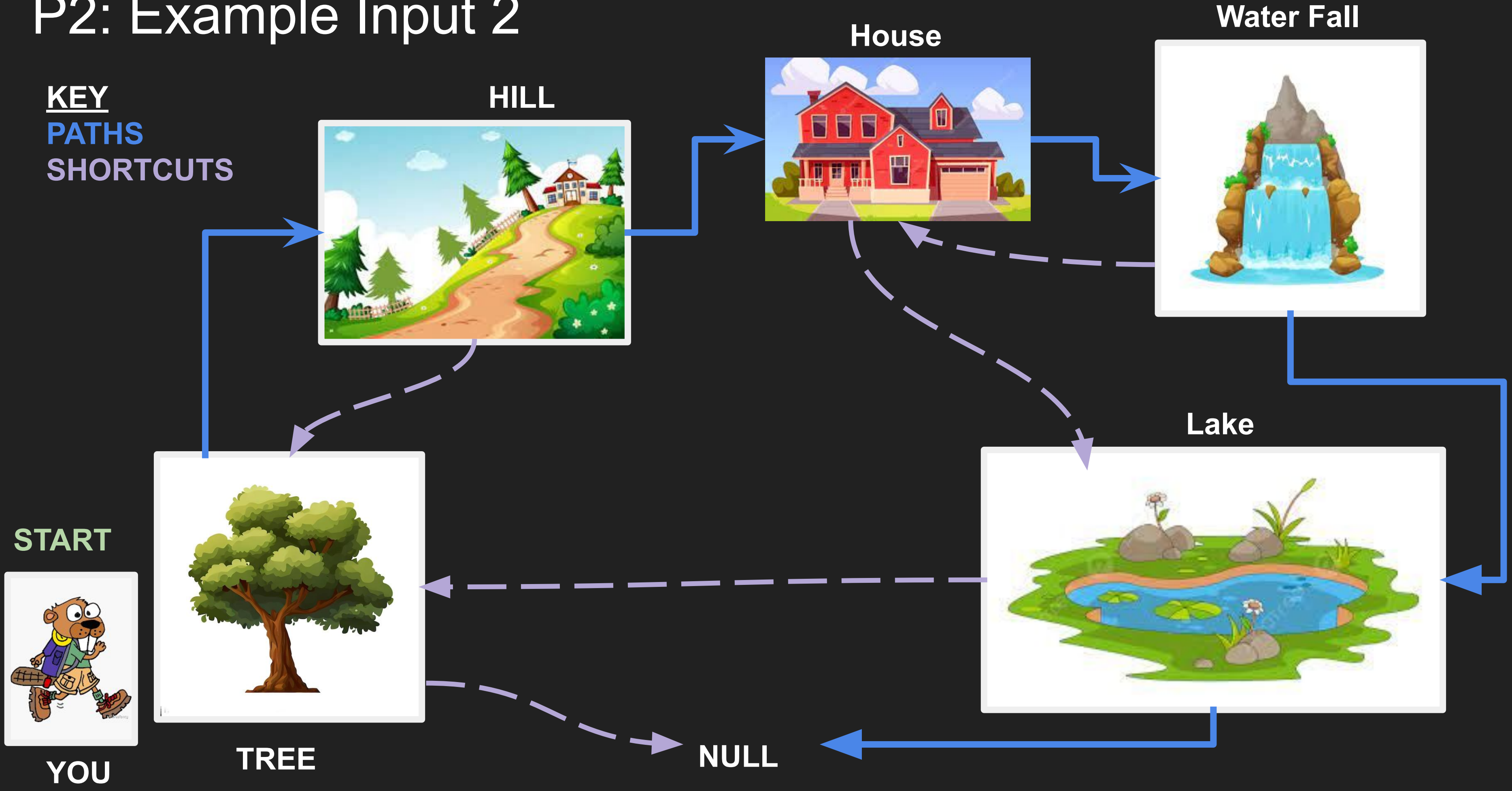
```
Node* copyHike(Node* start) {
    //algorithm
    //return start of copied hike
}
```

Example Node

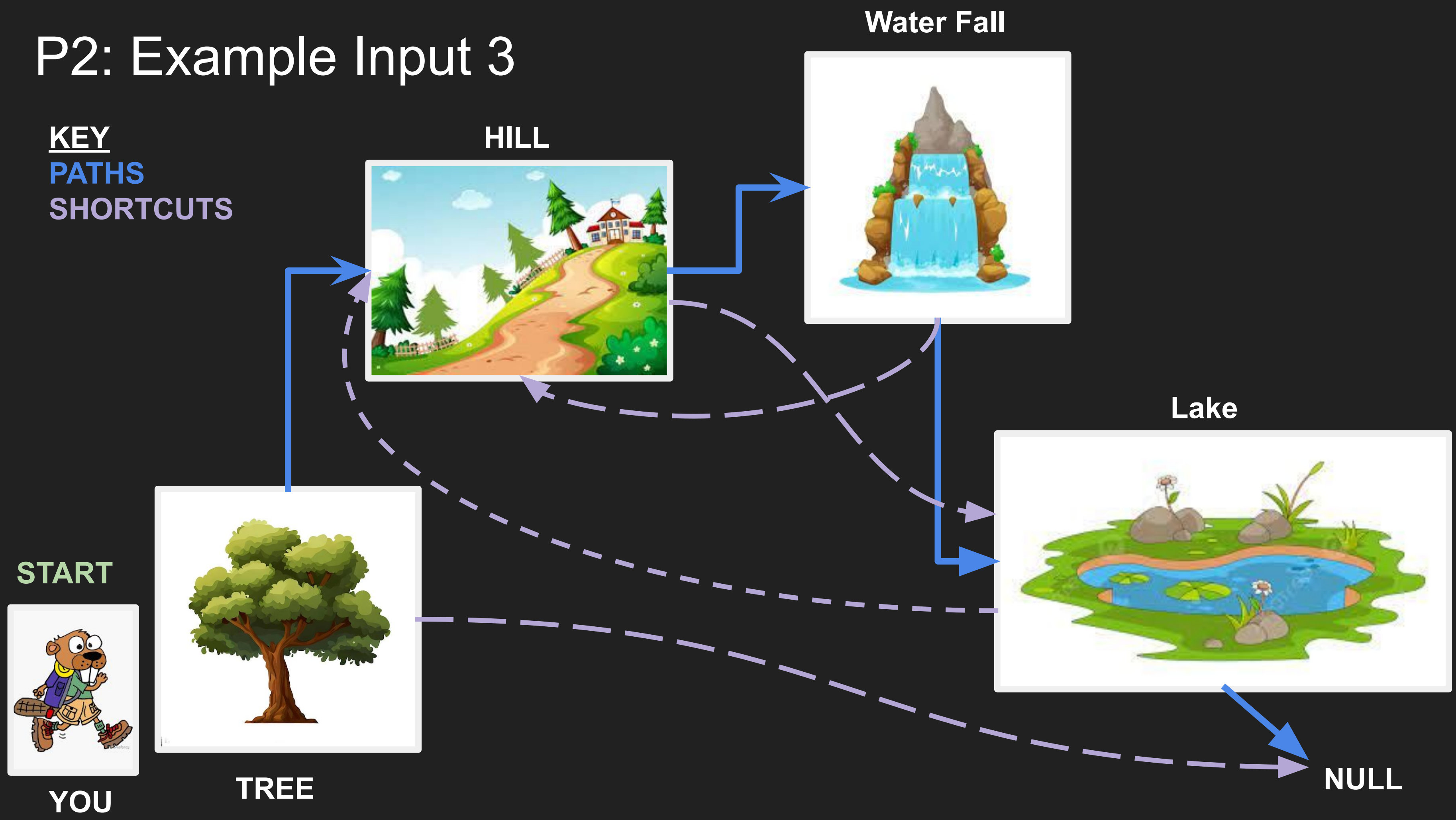
```
Node {
        Node* next;
        Node* Shortcut;
        String landmark;
};
```

P2: Example Input 2

KEY
PATHS
SHORTCUTS



P2: Example Input 3



SUBMIT

Upload video to coogTube or any other video sharing platform (youtube)

FF → Coogtube

Upload code to repl.it or any other code sharing platform

Fill out https://forms.gle/7rhu2E13wAcF2G6P8

SUBMIT **BEFORE** 9 PM

Contact me if there are any upload/submission errors

Submit even if unfinished. There is partial credit!