

## 582. Kill Process

Add to List

DescriptionHintsSubmissionsSolutions

- Total Accepted: **849**
- Total Submissions: **2121**
- Difficulty: **Medium**
- Contributors: [fallcreek](#)

Given  $n$  processes, each process has a unique **PID (process id)** and its **PPID (parent process id)**.

Each process only has one parent process, but may have one or more children processes. This is just like a tree structure. Only one process has PPID that is 0, which means this process has no parent process. All the PIDs will be distinct positive integers.

We use two list of integers to represent a list of processes, where the first list contains PID for each process and the second list contains the corresponding PPID.

Now given the two lists, and a PID representing a process you want to kill, return a list of PIDs of processes that will be killed in the end. You should assume that when a process is killed, all its children processes will be killed. No order is required for the final answer.

**Example 1:**

Input:

pid = [1, 3, 10, 5]

ppid = [3, 0, 5, 3]

kill = 5Output: [5,10]Explanation:

3

/ \

1 5

/

10

Kill 5 will also kill 10.

**Note:**

1. The given kill id is guaranteed to be one of the given PIDs.
2.  $n \geq 1$ .

[Subscribe](#) to see which companies asked this question.

Show Tags

```
class Solution {
public:
    vector<int> killProcess(vector<int>& pid, vector<int>& ppid, int kill)
    {
        vector<int> killed;
        map<int, set<int>> children;
        for (int i = 0; i < pid.size(); i++) {
            children[ppid[i]].insert(pid[i]);
        }
        killall(kill, children, killed);
        return killed;
    }

private:
    void killall(int pid, map<int, set<int>>& children, vector<int>& killed)
    {
        killed.push_back(pid);
        for (int child : children[pid]) {
            killall(child, children, killed);
        }
    }
};
```