Algorithms and Data Structures

Exercise Session 12

https://n.ethz.ch/~ahmala/and

Quiz

DP TASK

Given a list of n distinct positive integers, return the size of the largest subset such that for every pair of elements a and b in the subset, either a % b == 0 or b % a == 0.

Required Time Complexity: O(n^2)

Example: [4, 7, 8, 16, 28]

Answer: 3

dp[i]: size of the maximum subset that ends with i-th number.

```java
class Solution {
    public Integer largestDivisibleSubset(int[] nums) {
        int n = nums.length;
        Arrays.sort(nums);
        int[] dp = new int[n];
        for(int i = 0; i < n; i++) {
            dp[i] = 1;
        }
        for (int i = 0; i < n; i++) {
            for (int j = 0; j < i; j++) {
                if (nums[i] % nums[j] == 0) {
                    dp[i] = Math.max(dp[i], dp[j] + 1);
                }
            }
        }
        int ans = 0;
        for(int i = 0; i < n; i++) {
            ans = Math.max(ans, dp[i]);
        }
        return ans;
    }
}
```

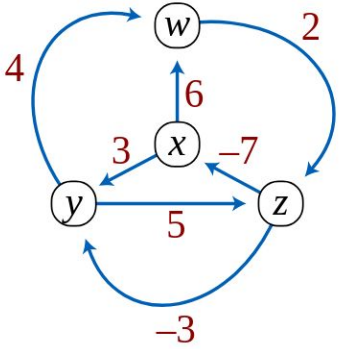
What if we wanna return the Elements?

```java
public List<Integer> largestDivisibleSubset(int[] nums) {
    int n = nums.length;
    Arrays.sort(nums);
    int[] dp = new int[n];
    int[] prev = new int[n];
    Arrays.fill(dp, 1);
    Arrays.fill(prev, -1);
    int maxIndex = 0;
    for (int i = 0; i < n; i++) {
        for (int j = 0; j < i; j++) {
            if (nums[i] % nums[j] == 0) {
                if(dp[i] < dp[j] + 1) {
                    dp[i] = dp[j] + 1;
                    prev[i] = j;
                }
            }
        }
        if (dp[i] > dp[maxIndex]) {
            maxIndex = i;
        }
    }
    List<Integer> result = new ArrayList<>();
    int index = maxIndex;
    while(index != -1) {
        result.add(nums[index]);
        index = prev[index];
    }
    return result;
}
```

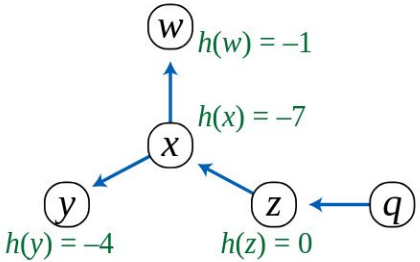
All Pairs Shortest Path Problem

- Johnson's Algorithm
- Floyd-Warshall

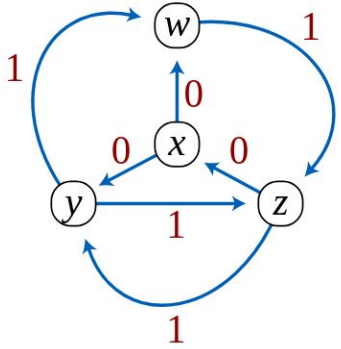
Johnson's Algorithm



original graph
with negative edges



shortest path tree
found by Bellman-Ford



reweighted graph with
no negative edges

Floyd-Warshall Algorithm

Peer Grading

Exercise 11.1