COMP4128 Week 05 Tutorial

Yifan He

z5173587@unsw.edu.au
`https://github.com/hharryyf/COMP4128-23T3-tutoring`

Reminder

- Contest 2, this weekend
- Topics: binary search, greedy, data structure, dp

Reminder

- Contest 2, this weekend
- Topics: binary search, greedy, data structure, dp
- Hints for Problem Set 4 by email (again)

Outline

- MST revision
- Shichikuji and Power Grid
- Ehab's Last Corollary
- A quick review for contest 2

Minimum Spanning Tree

Given an undirected weighted connected graph with V vertices and E edges. Each edge e_i is represented by a tuple (u_i, v_i, c_i) meaning this edge connects u_i and v_i with weight c_i. Pick a subset of edges of the graph so that this subset of edges can still make the graph connected. What is the minimum cost of the picked edges?

MST revision

Kruskal's algorithm

- Sort the edges in increasing order of weights
- Scan the edges one by one, if the edge creates a cycle, skip it, otherwise, add it to the graph
- The added edges form the MST of the graph
- Time complexity: $O(E \cdot log(E))$ with union-find

MST revision

MST properties

- For a graph G, suppose one of its MST has edge weights $w_1 \leq w_2 \leq ... \leq w_{V-1}$, another MST has edge weights $w'_1 \leq w'_2 \leq ... \leq w'_{V-1}$. Then, we have $w_1 = w'_1, w_2 = w'_2 ..., w_{V-1} = w'_{V-1}$.

MST revision

MST properties

- For a graph G, suppose one of its MST has edge weights $w_1 \leq w_2 \leq ... \leq w_{V-1}$, another MST has edge weights $w'_1 \leq w'_2 \leq ... \leq w'_{V-1}$. Then, we have $w_1 = w'_1, w_2 = w'_2 ..., w_{V-1} = w'_{V-1}$.
- If all the edges in the graph have different weights, the MST of the graph is unique

MST revision

MST properties

- For a graph G, suppose one of its MST has edge weights $w_1 \le w_2 \le ... \le w_{V-1}$, another MST has edge weights $w'_1 \le w'_2 \le ... \le w'_{V-1}$. Then, we have $w_1 = w'_1, w_2 = w'_2 ..., w_{V-1} = w'_{V-1}$.
- If all the edges in the graph have different weights, the MST of the graph is unique
- The MST that must contain a specific edge can be obtained by adding the edge to the MST of the original graph and removing the edge with the largest weight in the cycle created

Shichikuji and Power Grid

Given N cities ($N \leq 2,000$), each has coordinate (x_i, y_i). Building a power station at city i has cost c_i and connecting i and j costs $(k_i + k_j) \cdot (|x_i - x_j| + |y_i - y_j|)$. All cites must be connected to power. Calculate the minimum cost.

Shichikuji and Power Grid

Given N cities ($N \leq 2{,}000$), each has coordinate (x_i, y_i). Building a power station at city i has cost c_i and connecting i and j costs $(k_i + k_j) \cdot (|x_i - x_j| + |y_i - y_j|)$. All cites must be connected to power. Calculate the minimum cost.

Analysis

- If we only need to connect all cities together, it is just standard MST.

Shichikuji and Power Grid

Given N cities ($N \leq 2{,}000$), each has coordinate (x_i, y_i). Building a power station at city i has cost c_i and connecting i and j costs $(k_i + k_j) \cdot (|x_i - x_j| + |y_i - y_j|)$. All cites must be connected to power. Calculate the minimum cost.

Analysis

- If we only need to connect all cities together, it is just standard MST.
- But at least 1 city must connect to power directly

Shichikuji and Power Grid

Given N cities ($N \leq 2{,}000$), each has coordinate (x_i, y_i). Building a power station at city i has cost c_i and connecting i and j costs $(k_i + k_j) \cdot (|x_i - x_j| + |y_i - y_j|)$. All cites must be connected to power. Calculate the minimum cost.

Analysis

- If we only need to connect all cities together, it is just standard MST.
- But at least 1 city must connect to power directly
- How to solve this additional requirement?

Shichikuji and Power Grid

Analysis

- Create an additional vertex 0
- Connect 0 to city i with cost c_i

Shichikuji and Power Grid

Analysis

- Create an additional vertex 0
- Connect 0 to city *i* with cost c_i
- This forces us to pick at least 1 edge connecting vertex 0, which satisfies the “at least 1 city must connect to power directly” constraints

Shichikuji and Power Grid

Analysis

- Create an additional vertex 0
- Connect 0 to city i with cost c_i
- This forces us to pick at least 1 edge connecting vertex 0, which satisfies the “at least 1 city must connect to power directly” constraints
- The problem is now the standard MST problem
- Time complexity: $O(N^2 \cdot log(N))$

Shichikuji and Power Grid

Analysis

- Create an additional vertex 0
- Connect 0 to city i with cost c_i
- This forces us to pick at least 1 edge connecting vertex 0, which satisfies the “at least 1 city must connect to power directly” constraints
- The problem is now the standard MST problem
- Time complexity: $O(N^2 \cdot log(N))$
- This problem is a standard MST trick. It can be asked in programming interviews [a]

[a] `https://leetcode.com/problems/optimize-water-distribution-in-a-village/`

Ehab's Last Corollary

Given a connected undirected graph with n ($n \le 2e5$) vertices and an integer k, you have either:

- find an independent set that has exactly $\lceil \frac{k}{2} \rceil$ vertices.
- or find a simple cycle of length at most k.

An independent set is a set of vertices such that no two of them are connected by an edge. A simple cycle is a cycle that doesn't contain any vertex twice.

Ehab's Last Corollary

Analysis

- Shortest cycle in a graph: $O(|V| \cdot |E|)$
- Maximum independent set: NP-hard

Ehab's Last Corollary

Analysis

- Shortest cycle in a graph: $O(|V| \cdot |E|)$
- Maximum independent set: NP-hard
- No chance to solve it with brute force

Ehab's Last Corollary

Analysis

- Shortest cycle in a graph: $O(|V| \cdot |E|)$
- Maximum independent set: NP-hard
- No chance to solve it with brute force
- Simple case, what if the graph is a tree?

Ehab's Last Corollary

Analysis

- Shortest cycle in a graph: $O(|V| \cdot |E|)$
- Maximum independent set: NP-hard
- No chance to solve it with brute force
- Simple case, what if the graph is a tree?
- We can "bipartite" the graph, find the part with more vertices, and create an independent set of size K.

Ehab's Last Corollary

Analysis

- Shortest cycle in a graph: $O(|V| \cdot |E|)$
- Maximum independent set: NP-hard
- No chance to solve it with brute force
- Simple case, what if the graph is a tree?
- We can "bipartite" the graph, find the part with more vertices, and create an independent set of size K.
- What about the general case?

Ehab's Last Corollary

Analysis 2

- Consider the dfs tree of the graph, since the graph is undirected, we only have tree edges and back edges (i.e., no cross edges)

Ehab's Last Corollary

Analysis 2

- Consider the dfs tree of the graph, since the graph is undirected, we only have tree edges and back edges (i.e., no cross edges)
- We can extract all simple cycles with no edges cutting through

Ehab's Last Corollary

Analysis 2

- Consider the dfs tree of the graph, since the graph is undirected, we only have tree edges and back edges (i.e., no cross edges)
- We can extract all simple cycles with no edges cutting through
- There is at least 1 such cycle

Ehab's Last Corollary

Analysis 2

- Consider the dfs tree of the graph, since the graph is undirected, we only have tree edges and back edges (i.e., no cross edges)
- We can extract all simple cycles with no edges cutting through
- There is at least 1 such cycle
- If the cycle has size no more than K, print out the cycle

Ehab’s Last Corollary

Analysis 2

- Consider the dfs tree of the graph, since the graph is undirected, we only have tree edges and back edges (i.e., no cross edges)
- We can extract all simple cycles with no edges cutting through
- There is at least 1 such cycle
- If the cycle has size no more than K, print out the cycle
- Otherwise, create an independent set

Ehab's Last Corollary

Comments

- This problem examines the property of the dfs tree

Ehab's Last Corollary

Comments

- This problem examines the property of the dfs tree
- A **very** similar practice problem: Ehab's Last Theorem [a]

[a] `https://codeforces.com/contest/1325/problem/F`

Contest 2 Revision

Tips

- If finding min/max is too difficult, think if you can change the problem to a validation problem.

Contest 2 Revision

Tips

- If finding min/max is too difficult, think if you can change the problem to a validation problem.
- For dp problem: the size of input might be a hint

Contest 2 Revision

Tips

- If finding min/max is too difficult, think if you can change the problem to a validation problem.
- For dp problem: the size of input might be a hint

Additional practice problems

- Binary search: `https://codeforces.com/contest/985/problem/D`
- DP: `https://codeforces.com/contest/1227/problem/F1`
- Data structure: problem D in set 3