

Algorithm

Mid Term

Contest Hints

Same Component

Explanation:

আপনাকে একটি 2D গ্রিড দেওয়া হবে। যেখানে '.' আর '-' থাকবে। '.' দিয়ে বোঝায় আপনি সেই সেল এ যেতে পারবেন আর '-' দিয়ে বোঝায় আপনি সেই সেল এ যেতে পারবেন না। আপনাকে সোর্স আর ডেস্টিনেশন বলে দেওয়া হবে। আপনাকে বলতে হবে সোর্স থেকে ডেস্টিনেশনে যাওয়া পসিবল কিনা?

Hints

প্রথমে গ্রিড এর সারি আর কলাম সংখ্যা ইনপুট নিবেন। তারপর 2D গ্রিড ইনপুট নিবেন। তারপর সোর্স আর ডেস্টিনেশন ইনপুট নিয়ে মোডিউলে যেভাবে দেখানো হয়েছে সেভাবে সোর্স এর উপর DFS on 2D গ্রিড বা BFS অন 2D গ্রিড চালাবেন। তারপর দেখবেন ডেস্টিনেশন ভিজিটেড কিনা। যদি হয় তাহলে YES আর নাহলে NO প্রিন্ট করবেন।

Can Go?

Explanation:

আপনাকে N সংখ্যক নোড ও e সংখ্যক এজ দেওয়া থাকবে। সাথে প্রতি এজ এর সোর্স ডেস্টিনেশন ও কস্ট দেওয়া থাকবে। নোডগুলো ১ থেকে N অবধি থাকবে। আপনাকে একটা সোর্স দেওয়া থাকবে। তারপর টেস্টকেইস এ T সংখ্যক ডেস্টিনেশন আর DW কস্ট বলা থাকবে। আপনাকে বলতে হবে আপনি S থেকে D তে DW কিংবা তারও কম কস্ট এ যেতে পারবেন কিনা।

Hints

প্রথমে নোড আর এজ সংখ্যা ইনপুট নিবেন। তারপর গ্রাফটি ইনপুট নিবেন মোডিউলে যেভাবে দেখানো হয়েছে সেভাবে। তারপর সোর্স ইনপুট নিবেন। তারপর সোর্স বরাবর **Dijkstra** ফাংশন চালাবেন। এরপর টেস্টকেইস ইনপুট নিয়ে D আর DW ইনপুট নিবেন। তারপর ডিস্টেন্স গ্যারেতে D এর ডিস্টেন্স DW এর সমান বা ছোট কিনা চেক করবেন। যদি হয় তাহলে **yes** প্রিন্ট করবেন আর নাহলে **no** প্রিন্ট করবেন।

Can Go Again?

Explanation:

আপনাকে N সংখ্যক নোড ও e সংখ্যক এজ দেওয়া থাকবে। সাথে প্রতি এজ এর সোর্স ডেস্টিনেশন ও কস্ট দেওয়া থাকবে। নোডগুলো ১ থেকে N অর্ধি থাকবে। আপনাকে একটা সোর্স দেওয়া থাকবে। তারপর টেস্টকেইস এ T সংখ্যক ডেস্টিনেশন। আপনাকে বলতে হবে আপনি S থেকে D তে মিনিমাম কত কস্ট এ যেতে পারেন।

বি দ্রঃ যদি নেগেটিভ সাইকেল থাকে তাহলে Negative Cycle Detected প্রিন্ট করতে হবে।

Hints

প্রথমে নোড আর এজ সংখ্যা ইনপুট নিয়ে গ্রাফ এর এজলিস্ট ইনপুট নিবেন। তারপর ডিস্টেন্স এ্যারেতে ইনিশিয়ালি $1e18$ সেট করবেন। তারপর **Bellman ford** এলগোরিদম চালাবেন। যদি নেগেটিভ সাইকেল পান তাহলে Negative Cycle Detected প্রিন্ট করবেন। তারপর টেস্টকেইস ইনপুট নিয়ে সেটার লুপ চালাবেন আর ডেস্টিনেশন ইনপুট নিবেন। যদি সাইকেল না পান তাহলে চেক করবেন ডিস্টেন্স $1e18$ এর ছোট কিনা। যদি হয় তাহলে ডিস্টেন্স প্রিন্ট করবেন আর নাহলে Not Possible প্রিন্ট করবেন।

Shortest Distance

Explanation:

আপনাকে একটি গ্রাফের নোড N এবং edges E দেয়া হবে। Edges এর জন্য A , B এবং W দেয়া হবে, এইখানে A নোডের সাথে B নোডের edges আছে বুঝিয়েছে এবং এদের weight হচ্ছে W .

এরপর আপনাকে কিছু query দেয়া হবে। প্রতিটি query তে আপনাকে X এবং Y দেয়া হবে, এইখানে X হচ্ছে source এবং Y হচ্ছে destination। আপনার থেকে X নোড থেকে Y নোডে যেতে minimum cost বের করতে হবে। যদি X থেকে Y নোডে যাওয়া সম্ভব না হলে তখন - 1 প্রিন্ট করতে হবে।

দ্রষ্টব্য: এক নোড থেকে অন্য নোডে একাধিক edges থাকতে পারে।

Hints

এই প্রব্লেমটি আমরা **Floyd Warshall algorithm** দিয়ে করবো।
প্রথমে N এবং E ইনপুট নিব। এরপর একটি 2D অ্যারে নিব যেইখানে মডিউলে দেখানো উপায়ে infinity ($1e18$)

assign করবো।

এরপর **edges** গুলোকে ইনপুট নিব।

Edge ইনপুট নেয়া সময় খেয়াল রাখতে হবে যে, এক নোড থেকে অন্য নোডে একাধিক **edges** থাকতে পারে। যদি এক নোড থেকে অন্য নোডে একাধিক **edge** থাকে তখন সেইগুলোকে থেকে কম **cost** টিকে নিব।

এরপর মডিউলে দেখানো **Floyd Warshall algorithm** চালিয়ে দিব।

এরপর **query** ইনপুট নিব। প্রতিটি **query** তে পাওয়া **source** এবং **destination** এর **cost** টি প্রিন্ট করবো।

যদি **source** এবং **destination** এর **cost infinity** পাওয়া যায় তাহলে **-1** প্রিন্ট করবো , অন্যথায় **cost** প্রিন্ট করে দিব।

Knight Moves

Explanation:

এক্সাম ইনসট্রাকশন ভিডিওতে এই প্রশ্নটি ব্যাখ্যা করা হয়েছে, সেইটি দেখে ফেলুন।

Hints

প্রথমে টেস্ট কেস সংখ্যা ইনপুট নিয়ে নিতে পারেন। তারপর টেস্ট কেসের উপর লুপ চালিয়ে দিতে পারেন। প্রতিটি টেস্ট কেসে ছয়টি ইনপুট থাকবে। প্রথম দুটি হচ্ছে গ্রিডের সাইজ, পরের দুইটি হচ্ছে সোর্স এবং লাস্টের দুটি হচ্ছে ডেস্টিনেশন। ইনপুট নেওয়ার পর আপনি লুপ চালিয়ে সবগুলোর ডিসটেন্স -1 এবং ভিজিটেডকে ফলস করে দিতে পারেন। তারপর ইনপুটে থাকা সোর্স নোড এর উপর বিএফএস চালিয়ে দিতে পারেন। চালানোর পর ডেস্টিনেশন নোডের ডিসটেন্স টা প্রিন্ট করে দিলেই আউটপুট চলে আসবে।

