Handling shortest path tasks on Dense Graph with Segment Tree

Md Nafis Ul Haque Shifat

October 4, 2023

1 Introduction

কখনো কখনো দেখা যায় shortest path সম্পর্কিত সমস্যাগুলোতে এমন একটি গ্রাফ কন্সট্রাক্ট করতে হয় যেখানে একটি রেঞ্জে একটি নোড (কিংবা হয়তো নোডের অন্য কোনো রেঞ্জ) হতে এজ যোগ করতে হয় । এমন ক্ষেত্রে পুরো গ্রাফটি explicitly বানাতে গেলে কমপ্লেক্সিটি $O(n^2)$ হয়ে যায়। কখনো কখনো সেগমেন্ট ট্রি ব্যাবহার করে সেই গ্রাফটির equivalent একটি গ্রাফ কন্সট্রাক্ট করা যায়, যার উপর shortest path অ্যালগরিদম চালালেই হয় তখন। একটা সমস্যা দেখা যাক।

2 Problem (USACO '21 Tickets)

তোমার যাত্রাপথে মোট N $(1 \leq N \leq 10^5)$ টি চেকপয়েন্ট আছে। এছাড়াও পথে K $(1 \leq k \leq 10^5)$ টি টিকেট আছে, i-তম টিকেটটি c_i-তম চেকপয়েন্টে p_i দামে কিনা যাবে, এবং এই টিকেটটি থাকলে তুমি $[a_i, b_i]$ রেঞ্জের সকল চেকপয়েন্টে যেকোনো সময় যেকোনো চেকপয়েন্ট থেকে সরাসরি প্রবেশ করতে পারবে। সকল $i \in [1, n]$ এর জন্য, যদি তুমি শুরুতে i-তম চেকপয়েন্টে থাক, তবে 1 এবং N-তম চেকপয়েন্ট দুটিতে access পেতে সর্বনিম্ন কত খরচ করতে হবে?

Solution. প্রথমে সমস্যাটা গ্রাফ দিয়ে মডেল করার চেষ্টা করি। একটা সহজ উপায় হতে পারে আমরা প্রতি টিকেট এর জন্য নোড c_i থেকে i-তম টিকেটে p_i কস্ট এর একটি এজ দিব, আর i-তম টিকেট হতে $[a_i, b_i]$ রেঞ্জের সকল নোডে 0 কস্ট এর এজ দেই। যেমন ধরা যাক 5 টা চেকপয়েন্ট আছে, আর তিনটি টিকেট আছে, যেখানে (c_i, p_i, a_i, b_i) হচ্ছে যথাক্রমে $(1, 4, 2, 4), (4, 6, 1, 3), (3, 2, 4, 5)$; তাহলে গ্রাফ টা হবে নিচের মতো-

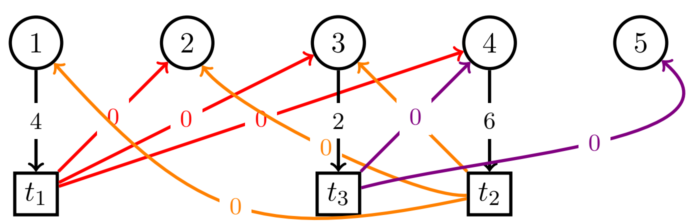

Figure 1

এখানে t_i দ্বারা i-তম টিকেট নোড কে নির্দেশ করা হয়েছে। আমাদের যদি i তম চেকপয়েন্ট থেকে শুরু করে 1 ও N উভয় চেকপয়েন্টে না পৌঁছে শুধু 1 নং চেকপয়েন্টে পৌঁছালেই হতো, তাহলে সমাধান কি হতো? একটু ভালো করে লক্ষ্য করলেই দেখবে তা হচ্ছে এই গ্রাফে i তম চেকপয়েন্ট থেকে 1 নং চেকপয়েন্টের শরটেস্ট পাথ! যদি i তম চেকপয়েন্ট থেকে 1 এবং N তম চেকপয়েন্টের পাথ disjoint হতো তাহলে আলাদা করে 1 ও N তম চেকপয়েন্টের জন্য শরটেস্ট পাথের কস্ট বের করে যোগ করে দিলেই হতো, সবসময় পাথদুটো disjoint হবে না। নোড s থেকে যাত্রা শুরু করলে অপটিমাল এন্সারে আমরা যেই এজ গুলো নিবো তাতে $s \rightsquigarrow 1$ পাথ এবং $s \rightsquigarrow N$ পাথ এ এমন একটি সাধারণ নোড z পাওয়া যাবে যেন উভয় পাথেই $s \rightsquigarrow z$ পর্যন্ত পাথটুকু সাধারণ থাকে এবং $z \rightsquigarrow 1$ এবং $z \rightsquigarrow N$ পাথ দুটোর এজ গুলোর সেট disjoint হয়। তবে আমাদের

টোটাল কস্ট দাঁড়াচ্ছে $D(s,z)+D(z,1)+D(z,N)$, যেখানে $D(u,v)$ হচ্ছে নোড u থেকে v এর শরটেস্ট পাথ কস্ট। এখন আমরা যদি কোনো ভাবে গ্রাফের সকল নোড z এর জন্য $D(z,1)$ ও $D(z,N)$ বের করতে পারি, তাহলে বাকি কাজটা বেশ সহজ হয়ে যায়, কেনোনা আমরা সকল নোড i এর কস্টকে $D(i,1)+D(i,N)$ দিয়ে initialize করে গ্রাফের এজ গুলোর ডিরেকশন উলটো করে দিয়ে সকল নোডকে source এ রেখে একটি Dijkstra চালালেই সকল নোডের জন্য উত্তর পেয়ে যাব! কারণ বুঝা খুব কঠিন না, লক্ষ্য কর, Dijkstra শেষে কোনো নোড s এর ফাইনাল উত্তর দাঁড়াচ্ছে সকল নোড z এর জন্য $D(z,1)+D(z,N)+D(z,s)$ এর মিনিমাম, ঠিক যেমনটা হওয়া উচিত।
কাজেই আমাদের এখন চ্যালেঞ্জ হচ্ছে সকল নোড s এর জন্য $D(s,1)$ এবং $D(s,N)$ বের করা। আমরা যদি আমাদের আগের গ্রাফের সবগুলো এজ রিভার্স করে দেই তাহলে 1 আর N থেকে আলাদা করে দুটি Dijkstra চালালেই কিন্তু আমরা আমাদের উত্তর পেয়ে যাচ্ছি। কিন্তু সমস্যা হচ্ছে আমাদের গ্রাফ এ $O(N^2)$ টা এজ থাকতে পারে। কিন্তু এজগুলো যেহেতু $O(N)$ টি রেঞ্জ ফর্ম করে, তাই সেগমেন্ট ট্রি কাজে লাগিয়ে আমরা অপটিমাইজ করতে পারব।
মুল গ্রাফে আমরা i-তম টিকেট থেকে $[a_i,b_i]$ রেঞ্জের সকল চেকপয়েন্টে একটি করে 0 ওয়েট এর এজ দিচ্ছিলাম। এখন আমরা যা করব তা হচ্ছে- সবগুলো চেকপয়েন্ট গুলোর উপর একটি সেগমেন্ট ট্রি বানাব, আর সেগমেন্ট ট্রি তে ওই রেঞ্জের অন্তর্ভুক্ত যে $O(\log n)$ টা নোড আছে তাতে i-তম টিকেট হতে 0 কস্টের এজ দিব। একই সাথে সেগমেন্ট ট্রি তে সকল প্যারেন্ট নোড হতে চাইল্ড নোডেও 0 কস্টের এজ দিব। এখানের সবগুলো এজ কিন্তু ডিরেক্টেড।

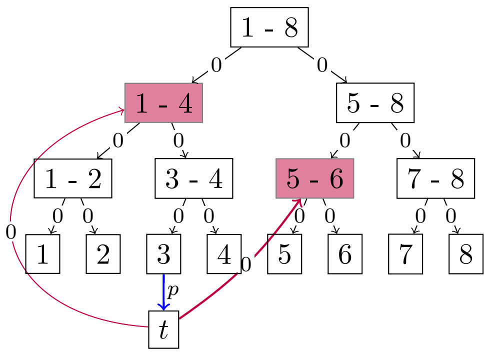

Figure 2: যখন $[a,b]=[1,6]$

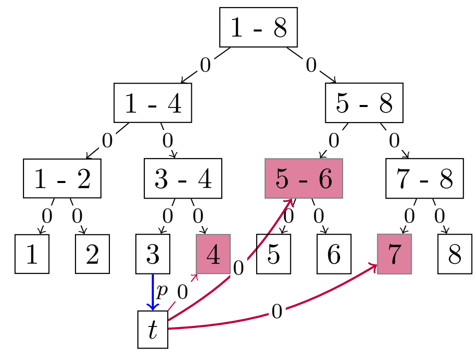

Figure 3: যখন $[a,b]=[4,7]$

একটু ভালো করে লক্ষ্য করলেই বুঝবে মূল গ্রাফে যদি কোনো টিকেট t হতে কোনো চেকপয়েন্টে 0 কস্টের এজ থাকে, তবে আমাদের এই নতুন গ্রাফেও টিকেট t হতে ওই চেকপয়েন্টে 0 কস্টের একটি পাথ আছে। কেনোনা যদি t থেকে কোনো নোড u তে এজ থেকে থাকে, তাহলে সেগমেন্ট ট্রি তে এমন একটি নোড $[l,r]$ পাওয়া যাবে যেন $u\in[l,r]$ এবং t থেকে নোড $[l,r]$ এ একটি এজ 0 কস্টের এজ আছে। আবার যেহেতু $[l,r]$ থেকে তার নিচের সকল নোড 0 কস্টে reachable, তাই আসলে t ও নোড u এর মধ্যে অবশ্যই 0 কস্টের পাথ থাকতে হবে। তার মানে এই নতুন গ্রাফে যেকোনো দুটি নোডের শরটেস্ট পাথের কস্ট আর আগের গ্রাফের যেকোনো দুটি নোডের শরটেস্ট পাথের কস্ট একই। তবে যেহেতু আমাদের 1 আর N হতে সকল নোডের কস্ট লাগবে, তাই আমরা এই গ্রাফটির সব এজ রিভার্স করে 1 আর N হতে Dijkstra চালালেই উত্তর পেয়ে যাবো!গ্রাফে যেহেতু $O(N\log N)$ টি এজ আছে, আবার Dijkstra একটা নতুন $\log n$ ফ্যাক্টর যোগ করবে, তাই ফাইনাল কমপ্লেক্সিটি দাঁড়াচ্ছে $O(N\log^2 N)$। আমার কোড এখানে দেখতে পার।

এই সমাধানটা বুঝলে এবার আমরা আরেকটা সমস্যা দেখতে পারি।

3 Problem (CodeChef Dense Graph)

একটি N নোডের unweighted ডিরেক্টেড গ্রাফ রয়েছে, সাথে তোমাকে $2M$ টা রেঞ্জ - $[A_1,B_1],[A_2,B_2],\ldots,[A_m,B_m]$ এবং $[C_1,D_1],[C_2,D_2],\ldots,[C_m,D_m]$- দেয়া আছে। সকল $i\in[1,m]$ এর জন্য, সকল $u\in[A_i,B_i]$ এবং $v\in[C_i,D_i]$ এর জন্য, u থেকে v তে একটি ডিরেক্টেড এজ আছে। এছাড়াও তোমাকে দুটো নোড X এবং Y দেয়া থাকবে, X থেকে Y তে যেতে শরটেস্ট ডিস্টেন্স বের করতে হবে।

Solution. এই সমস্যার সাথে আগের সমস্যার একটি মূল পার্থক্য হচ্ছে এবার একটি রেঞ্জ থেকে আরেকটি রেঞ্জ এর নোড গুলোর মধ্যে এজ যাচ্ছে। স্বাভাবিকভাবেই এবার আমরা সেগমেন্ট ট্রি তে $[A_i,B_i]$ এর মধ্যে থাকা নোড গুলো থেকে

$[C_i, D_i]$ রেঞ্জের নোড গুলোর মধ্যে এজ টানবো, কিন্তু একটু সমস্যা আছে। লক্ষ্য কর সেগমেন্ট ট্রি তে $[A_i, B_i]$ রেঞ্জের যে নোড গুলো থেকে এজ দিব তাদের চাইল্ড গুলো থেকে সেসব নোড reachable হতে হবে, কেনোনা সেগমেন্ট ট্রি তে $[l, r]$ নোড থেকে এজ যাওয়ার অর্থ হচ্ছে l হতে r পর্যন্ত সকল নোড হতে এজ যাওয়া, তাই লিফ নোড গুলো হতে অবশ্যই $[l, r]$ নোড টি reachable হতে হবে। কিন্তু একই ভাবে আবার সেগমেন্ট ট্রি তে $[C_i, D_i]$ রেঞ্জের মধ্যে থাকা যে নোড গুলোতে এজ দেয়া হবে সেসব নোড থেকে তাদের চাইল্ড নোড গুলো অবশ্যই reachable হতে হবে। তাহলে আমরা সেগমেন্ট ট্রির এজ গুলো কোন ডিরেকশনে দিব? সমাধান হচ্ছে আমরা দুটি সেগমেন্ট ট্রি নিবো, কিন্তু এদের leaf নোড গুলো সাধারণ হবে, আর ট্রি দুটিতে এজের ডিরেকশন আলাদা হবে (Figure 4 লক্ষ্য কর)। প্রথম সেগমেন্ট ট্রি তে $[A_i, B_i]$ রেঞ্জের মধ্যে থাকা নোড গুলো থেকে দ্বিতীয় সেগমেন্ট ট্রি তে $[C_i, D_i]$ রেঞ্জে থাকা নোড গুলোর মধ্যে একটি এজ দিব, যাদের weight হবে 1। এবার এই গ্রাফে নোড X থেকে 0-1 BFS চালিয়ে নোড Y এর শরটেস্ট ডিস্টেন্স নিলেই আমরা উত্তর পেয়ে যাবো! কিন্তু এতে প্রতি রেঞ্জের জন্য $O(\log^2 n)$ টা এজ লাগবে; আমরা এটিকে সহজেই $O(\log n)$ করতে পারি- আমরা একটি dummy নোড T নিবো, এবং প্রথম সেগমেন্ট ট্রির নোড গুলো থেকে 1 weight এর এজ T তে দিব এবং T থেকে 0 weight এর এজ দ্বিতীয় সেগমেন্ট ট্রির নোড গুলোতে দিব। যদি $[A_i, B_i] = [6, 8]$ এবং $[C_i, D_i] = [1, 5]$ হয়, তবে গ্রাফটি এমন হতে পারে-

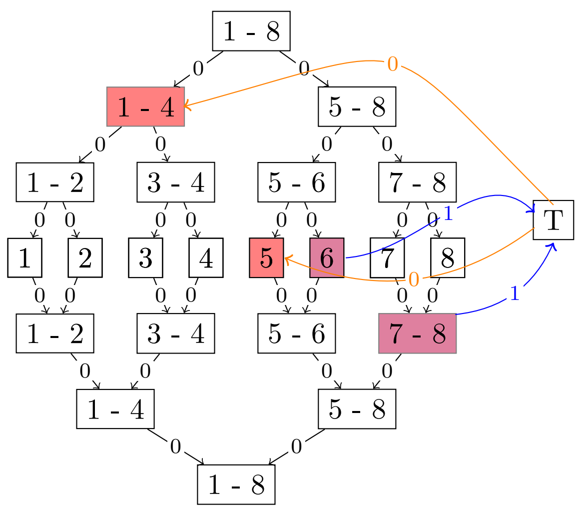

Figure 4

এখন কথা হচ্ছে এটি কেনো কাজ করে? এবারো আসলে দেখানো যায় যদি মূল গ্রাফে দুটো নোড u থেকে v তে একটি 1 weight এর এজ থেকে থাকে, তাহলে আমাদের এই নতুন গ্রাফেও u থেকে v তে একটি 1 কস্টের পাথ আছে । প্রমাণ টাও সহজ- u থেকে v তে এজ থাকলে, নিচের সেগমেন্ট ট্রি তে এমন একটি নোড $[l_1, r_1]$ পাওয়া যাবে যেন $u \in [l_1, r_1]$, কাজেই u থেকে 0 কস্টে $[l_1, r_1]$ reachable এবং উপরের ট্রি তে এমন একটি নোড $[l_2, r_2]$ পাওয়া যাবে যেন $v \in [l_2, r_2]$, কাজেই $[l_2, r_2]$ থেকে 0 কস্টে v reachable, এবং $[l_1, r_1]$ থেকে T তে 1 কস্টের ও T থেকে $[l_2, r_2]$ তে 0 কস্টের এজ আছে। তারমানে আমরা u থেকে v তে একটি 1 কস্টের পাথ পেয়ে যাচ্ছি। আমার কোড এখানে দেখতে পার।