

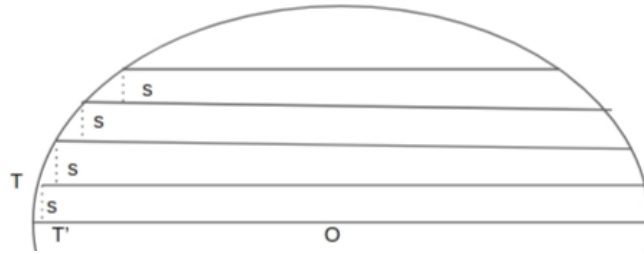
SCPC Kontes Mingguan 2

August 23, 2020

1 Gudang Kardus

Tag: Math

Perhatikan kita bisa membagi-bagi lingkaran menjadi potongan-potongan independen. Setiap potongan dapat diisi dengan kardus sebanyak mungkin dan tidak mempengaruhi potongan-potongan lain:



Anggap: O sebagai titik pusat lingkaran, T sebagai titik dimana garis berjarak S dan sejajar dengan diameter memotong lingkaran, T' sebagai proyeksi T pada diameter. Kita berusaha mencari panjang OT'. ini dapat dicari dengan rumus $\sqrt{R^2 - S^2}$ karena $OT = R$ dan $TT' = S$. Setelah menemukan panjang OT', banyaknya kardus yang dapat dimasukkan pada lapisan ini adalah $\frac{2OT'}{S}$

Dengan logika yang sama, panjang daerah yang dapat diletakkan kardus pada lapisan ke-i adalah $\sqrt{R^2 - iS^2}$. Jadi, lakukan iterasi selama iS^2 masih lebih kecil dari R , lalu tambahkan semua jawabannya. Mungkin terdapat masalah presisi, sehingga kamu bisa mencari

$$d = \frac{4(R^2 - iS^2)}{S^2}$$

Hasilnya kemudian adalah bilangan terbesar u yang memenuhi $u^2 \leq d$ menggunakan binary search.

Kompleksitas: $O(R \log R)$

2 Jus Tropis

Tag: Convex hull

Kita bisa memandang jus tropis dengan X pisang dan Y mangga sebagai titik (X, Y) pada koordinat kartesius. Selain itu, pertanyaan apakah bisa membuat jus dengan A mangga dan B pisang juga dipandang sebagai titik (A, B) pada koordinat kartesius.

Teorema: (A, B) Dapat dibuatkan jus tropis jika dan hanya jika titik (A, B) berada di dalam sebuah segitiga sembarang yang dibentuk oleh 3 titik (X, Y). Bukti diserahkan sebagai latihan kepada pembaca.

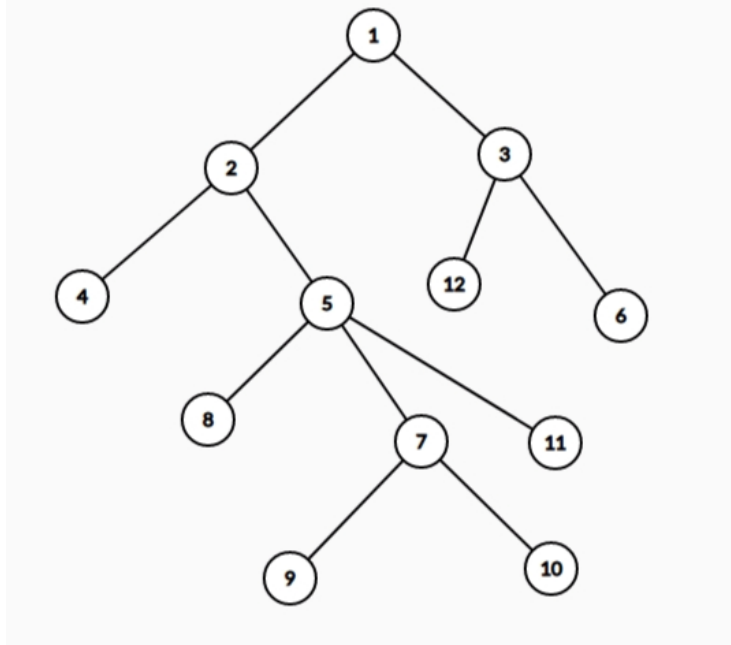
Jadi, salah satu solusinya adalah mencoba-coba semua kemungkinan segitiga lalu cek apakah (A, B) berada di dalamnya. Sayangnya solusi ini bekerja dalam $O(N^3)$. Untuk mendapatkan solusi $O(N)$, convex hull pada titik-titik dapat dicari lalu tinggal cek apakah (A, B) berada di dalam convex hull. Salah satu cara untuk melakukan pengecekan adalah cek apakah total sudut yang dibentuk oleh (A, B) dan sisi-sisi convex hull = 360^0 . Untuk membuat convex hull, kamu bisa menggunakan algoritma jarvis $O(N^2)$ atau graham scan $O(N \log N)$

Kompleksitas: $O(N^2)$

3 Pulau Sederhana

Tag: Combinatorics, Tree

Pertama sadar bahwa graph disini pasti berbentuk tree. Salah satu properti dari tree: dua pasang kota pasti memiliki tepat 1 jalan tercepat. Sehingga, kita bisa menghitung kontribusi dari setiap jalan, yaitu, berapa banyak pasang (i, j) yang menggunakan jalan tersebut. Untuk mempermudah, tree kita root di 1.



Sebut $SZ[i]$ sebagai banyaknya kota yang berada pada subtree dari i termasuk dirinya sendiri. Pada gambar diatas, $SZ[5] = 6$ (5, 7, 8, 9, 10, 11), $SZ[12] = 1$, $SZ[2] = 8$. Anggap kita mau mencari kontribusi dari jalan yang menghubungkan 2 dan 5. Pasangan yang valid pasti salah satu kota berada di subtree 5, dan satu kota lain berada diluar subtree 5. Dengan kata lain, kontribusi jalan (2, 5) ke jawaban adalah $SZ[5] * (N - SZ[5])$. Jadi, hitung semua SZ lalu hitung kontribusi dari masing-masing jalan. Perhitungan SZ dapat dilakukan dalam $O(N)$ menggunakan dynamic programming, yaitu:

$$SZ[i] = 1 + \sum_{j \in anak(i)} sz[j]$$

Kompleksitas: $O(N)$

4 Negara Chanek

Tag: Minimum Spanning Tree

Karena bobot graphnya yang unik, jalan ke- i , yang menghubungkan kota (A, B), pasti dipakai jika dan hanya jika belum ada rute yang menghubungkan kota (A, B) dengan jalan ke-1 sampai ke- $(i-1)$. Jadi, kamu bisa membangun MST menggunakan algoritma kruskal dengan mengambil secara terurut mulai dari edge pertama. Lalu setelah MST dibentuk, kamu bisa menggunakan metode sama seperti soal "Pulau Sederhana" untuk menghitung total jalan untuk masing-masing pasang kota.

Kompleksitas: $O(M \log M)$

5 Memotong Kue

Tag: Pola, OEIS

Kamu bisa mencoba-coba untuk nilai-nilai N kecil lalu mencari pola. Untuk $N = 1$, ada 6 daerah. N lebih tinggi: <http://www.randomwalk.de/sequences/a091908.pdf>.

Setelah menemukan 3 suku pertama: buka oeis.org, masukkan beberapa elemen pertama, lalu dapatkan rumusnya :)

Kompleksitas: $O(1)$

6 Distortion World

Tag: BFS

Solusi pertama yang terpikirkan adalah BFS dengan state (x, y, z) sekarang. Namun solusi ini akan kena memori limit karena batas memori yang ketat. Untuk mengoptimisasi, perhatikan bahwa paling banyak ada $6N^2$ states. Sehingga dibanding menyimpan state dalam array `visited[x][y][z]`, kamu bisa menyimpannya dalam sebuah map sehingga hanya state-state yang dikunjungi yang tersimpan.

Kompleksitas: $O(N^2)$

7 Euis

Tag: Combinatorics

Perhatikan setiap pembacaan kata independen, sehingga kita bisa menghitung banyaknya kemungkilan eu (sebut x) dan e (sebut y). Jawabannya x^3y^2 .

Kompleksitas: $O(\|S\|)$

8 Odd Bubble Sort 1

Tag: Simulation

Untuk mendapatkan array awal, kamu bisa mensimulasikan secara terbalik dari array sorted sehingga mendapatkan array awal dengan melakukan langkah mulai dari langkah terakhir. Sebagai contoh pada sample pertama testcase pertama:

1. Awal mulai dengan array sorted [1, 2, 3, 4, 5]
2. saat $i = 5$, yang tetap adalah 2, 3, 4, 5. Sehingga array tetap
3. saat $i = 4$, yang tetap adalah 3, 4, 5. Sehingga elemen pertama dan kedua ditukar. Array menjadi [2, 1, 3, 4, 5]
4. Saat $i = 3$, yang tetap adalah 4, 5. Sehingga elemen pada indeks 2 dan 3 ditukar. Array menjadi [3, 2, 1, 4, 5]. Perhatikan urutan penukaran mulai dari indeks terbesar.
5. Saat $i = 2$, yang tetap adalah 2, 5. Sehingga array menjadi [3, 4, 2, 1, 5].
6. Saat $i = 1$, yang tetap adalah 3. Sehingga array menjadi [4, 3, 5, 2, 1]

Kompleksitas: $O(N^2)$