Zadanie H: Hyperloop

Limit czasowy: 50s, limit pamięciowy: 64MB.

Jest rok 2077. Sztuczna inteligencja rozwiązała właśnie ostatni problem kolorowania hipergrafów, a ekscentryczny miliarder Melon Usk zlecił świeżo pozbawionym zatrudnienia pracownikom Informatyki Analitycznej stworzenie oprogramowania dla kompleksu Hyperloop – systemu superszybkich połączeń międzymiastowych.

Z powodu rosnącego poziomu wód, jedyną masą lądową pozostałą na Ziemi jest pojedyncza wyspa, na której brzegu leży n ocalałych miast, nazwanych dla uproszczenia 1, 2, ..., n w kolejności występowania wzdłuż brzegu. Melon jest właścicielem promów kursujących wokół wyspy, a więc pomiędzy miastami i i $i+1$ dla $1 \leqslant i < n$ oraz pomiędzy miastami n i 1, i połączeń Hyperloop, łączących pewne pary *niesąsiednich* miast [1]. Pozostało jedynie zaprojektowanie oprogramowania wyznaczającego najkrótsze trasy pomiędzy miastami; w wersji beta będzie ono obsługiwać jedynie podróż z miasta 1 do miasta n.

Zadanie mogłoby zdać się trywialne: wszak oprogramowanie ma brać pod uwagę oba rodzaje połączeń, a między miastami 1 i n istnieje już *bezpośrednie* połączenie za pomocą promu, jednak wcale nie musi być ono najszybsze! Do tego może istnieć wiele najkrótszych ścieżek. W takiej sytuacji system powinien preferować ścieżki, w których skład wchodzą jak najdłuższe połączenia (im dłużej trwa pojedyncze połączenie, tym pasażer może być bardziej skłonny do zamówienia w jego trakcie kawy, pomnażając majątek Uska). Formalnie, aby porównać ścieżki **o tej samej sumarycznej długości**, należy wypisać długości występujących w nich połączeń (uwzględniając powtórzenia), posortować każdy z ciągów nierosnąco, i wybrać najpóźniejszy leksykograficznie [2].

Mając dany opis promów kursujących wokół wyspy oraz połączeń Hyperloop, znajdź optymalną ścieżkę z miasta 1 do miasta n. Podczas porównywania ścieżek oba rodzaje połączeń traktowane są w taki sam sposób, a wszystkie połączenia są dwukierunkowe.

Zwróć uwagę, że to zadanie ma niski limit pamięci – 64MB.

Wejście

Pierwsza linia wejścia zawiera liczbę zestawów danych z ($1 \leqslant z \leqslant 600$). Potem kolejno podawane są zestawy w następującej postaci:

Pierwsza linia zestawu zawiera dwie liczby całkowite n oraz m ($3 \leqslant n \leqslant 100\,000$, $n \leqslant m \leqslant 300\,000$) – liczbę miast oraz liczbę połączeń.

Każda z kolejnych m linii zestawu zawiera trzy liczby całkowite u_i, v_i, d_i ($1 \leqslant u_i \neq v_i \leqslant n$, $1 \leqslant d_i \leqslant 50\,000$), oznaczające kolejno: numery miast które łączy dane dwukierunkowe połączenie, oraz czas potrzebny na jego przebycie.

Wśród połączeń podanych na wejściu pojawią się $(1, 2)$, $(2, 3)$, ..., $(n, 1)$, odpowiadające połączeniom używającym promu. Każdą parę miast łączyć będzie co najwyżej jedno połączenie.

Sumaryczna liczba miast oraz liczba połączeń we wszystkich przypadkach testowych nie przekroczy odpowiednio 400 000 i 800 000.

[1] Połączenia Hyperloop używają podziemnych tuneli. Tunele, które mogłyby się przeciąć, zostały zwyczajnie poprowadzone na różnych głębokościach.

[2] Ciąg $a_1, ..., a_p$ jest wcześniejszy leksykograficznie od ciągu $b_1, ..., b_q$ jeśli jest jego prefiksem lub istnieje takie $i \leqslant \min(p, q)$ że $a_j = b_j$ dla $j < i$ oraz $a_i < b_i$. W tym zadaniu porównywane ciągi mają zawsze tę samą sumę, a więc podczas porównywania dwóch różnych ciągów żaden z nich nie będzie prefiksem drugiego.

Wyjście

Dla każdego zestawu danych, w pierwszej linii wypisz jedną liczbę całkowitą k, oznaczającą liczbę miast na optymalnej ścieżce. W drugiej linii wypisz k różnych liczb całkowitych, oznaczających numery kolejnych miast. Wyznaczona ścieżka powinna zaczynać się w mieście 1, kończyć w mieście n, i być optymalna według definicji z treści zadania. Jeśli istnieje wiele optymalnych ścieżek, możesz wypisać dowolną z nich.

Przykład

Dla danych wejściowych:	Możliwą poprawną odpowiedzią jest:
2 4 6 1 2 1 1 3 2 2 3 1 2 4 2 3 4 1 1 4 4 6 11 1 2 9 2 3 12 3 4 3 4 5 5 5 6 10 6 1 22 2 4 9 3 6 1 4 6 5 2 5 2 3 5 8	3 1 2 4 5 1 2 5 3 6

Wyjaśnienie

W pierwszym teście przykładowym minimalna odległość z miasta 1 do miasta 4 wynosi 3, i realizują ją trzy ścieżki: $1 \to 2 \to 4$, $1 \to 3 \to 4$ i $1 \to 2 \to 3 \to 4$. Według definicji z treści zadania pierwsze dwie ścieżki są lepsze od trzeciej, bo w porządku leksykograficznym $(2, 1) > (1, 1, 1)$. Dowolna z dwóch pierwszych ścieżek jest poprawną odpowiedzią. Ścieżka bezpośrednia $1 \to 4$ nie jest rozważana, gdyż ma długość 4.

W drugim teście przykładowym minimalną odległość realizują ścieżki $1 \to 2 \to 5 \to 3 \to 6$ i $1 \to 2 \to 5 \to 4 \to 3 \to 6$. Optymalna jest pierwsza z nich, gdyż porównując ich posortowane ciągi odległości dostajemy $(9, 8, 2, 1) > (9, 5, 3, 2, 1)$.