

Анализ на задача travel

Тагове: BFS, моделиране/разширяване на графа, фиксиране на отговор

Решение на първа подзадача - 20 точки

Ограничението е достатъчно малко, за да може да минат всякакви идеи тип ”пълно изчерпване”.

Решение на втора подзадача - 50 точки

Впечатление прави, че $K = 1$, в тази подзадача, въпреки че $K \leq 2$ за цялата задача. Това предполага, че решението ни някак ще се възползва от ограничението - наистина можем да фиксираме “самолета”, който ще ползваме в пътя си от връх 1 към връх N . Да кажем, че сме фиксирали ребро $x - y$ - тогава пътят ни ще е един от два типа:

- $1 - x - y - N$
- $1 - y - x - N$

Чрез BFS ще намерим най-кратките пътища от 1 до x и y и от N до x и y . Тези резултати ще са ни нужни, за да изчислим най-кратките пътища от гореспоменатите два типа. Отговорът на задачата се явява най-краткият път от двата типа за всички ребра тим “самолет”. Реализацията на този подход е във файла **travel_m2_k=1.cpp**.

Решение на трета подзадача - 90 точки

Ще продължим да разширяваме идеята от миналата подзадача - все пак единственото, което се променя тук, е че имаме повече върхове в графа. Миналото ни решение беше $O(NM)$, така че определено има какво да се иска като оптимизации. Ще забележим, че постоянно преизчисляваме информация, която вече сме получили по-рано - за всяко ребро наново пускаме BFS от върхове 1 и N . Това видимо е ненужно и ще се оправим с пускането на тези обхождания веднъж - нужните разстояния ще запазим в два масива и при нужда ще ги взимаме от там. Сложността на алгоритъта пада на $O(N)$. Имплементацията е във файл **travel_m_k=1.cpp**

Решение на цялата задача - 100 точки

Пълното решение на задачата вече ни задължава да разрешим случая, при който $K = 2$ и това не става аналогично на миналите две подзадачи. Това е умишлено решение, за да могат да се изкарват повече точки на задачата.

Тук ще ни е полезна идеята за “разширяване” на граф. В някои задачи е полезно освен текущия връх да знаем и допълнителна информация (каква точно, зависи от задачата). Тук можем да пуснем BFS от 1 и освен да пазим текущия ни връх да поддържаме информация за колко “полета” сме ползвали дотук. Така всеки връх ще създаде три копия - ако е ползвал 0/1/2 полета. Сложността ще стане $O(NK)$, което е разумно за тази задача.

Допълнителни материали за “разширение на графи”:

- [въвеждащ блог за темата](#)
- [Зимен 2012 B3](#)
- [НОИ-3 2019 C6](#)
- [Open Moscow Final 2023 Day 2 Problem D](#)
- [НОИ-3 2023 A4](#)

Автор: Виктор Кожухаров Анализ: Иван Лупов