Графи

Качила съм отделно файл с имена на книги и линкове към файловете им.

Различни видове графи:

- неориентирани
- ориентирани
- прости
- мулти

} за ребра

- претеглени (ребра и/или върхове)
- непретеглени (ребра и/или върхове)

$G = \langle V, E \rangle$ ← м-во от ребра $E \subseteq V \times V$.

↑ м-во от върхове

Нека $\overline{\overline{V}} = n$ и $\overline{\overline{E}} = m$.

Представяния:

а) Матрица на съседство $A = (a_{ij})_{n \times n}$.

$$a_{ij} = \begin{cases} 1, & \langle i, j \rangle \in E \\ 0, & \text{else} \end{cases}$$

$M(n) = O(n^2)$

Ползва се за dense граф т.е. с поне $m = \Omega(n^2)$ ребра.

Иначе за sparse граф $m = O(n)$, то ползваме

б) Списъци на съседство.

$Adj[1 \ldots n]$

$Adj(v)$, $v \in V$

↳ списък на ребрата излизащи от връх v.

$M(n) = O(m) = O(n^2)$.

* Ориентирани графи има $0 \le m \le n \cdot (n-1)$ ребра
* Неориентирани: $0 \le m \le \frac{n(n-1)}{2}$
* Свързани граф: $n-1 \le m \le$ зависи от графа.

① Имаме ги обяви за/уп човека от типа „имам X, търся Y".
Вие имате предмет A и търсите предмет B. Как да си набавите B с минимален брой размени.
Имаме ориентиран граф
• върхове: предметите
• ребра: посока на размяната т.е. от връх u има ребро към връх v, ако някой има v и търси u.
Най-къс път по # ребра -> BFS.

② В едно сграда има k асансьора, всеки вози определен брой етажи $a_1, \dots, a_k$ (a_i - # етажи, които вози)
Как от етаж №1 да стигнем до последният етаж n.
a) с мин. # прехвърляния от асансьор на друг
b) за най-кратко време?

Неориентиран е графа (и нагоре и надолу може да се вози).

Върховете са етажите (и номер), а ребро м/у два етажа u и v има, ако някой асансьор вози от етаж u до етаж v (и на брой ребро).

а) Целта е по пътя да минимизираме т.е. брой ребра -> BFS -> $O(m+n)$.

б) Целта е да минимизираме време, все едно ще закачим на ребрата тегло колко етажа возят (положителни тегла). Тогава веча с Dijkstra за $O(m+n\lg n)$ е пирамида на Фибоначи намираме.

u $|v-u|$ v

③ Имаме неор. свързан граф. Да се намери път обхождащ всички върхове в двете посоки всяко ребро.

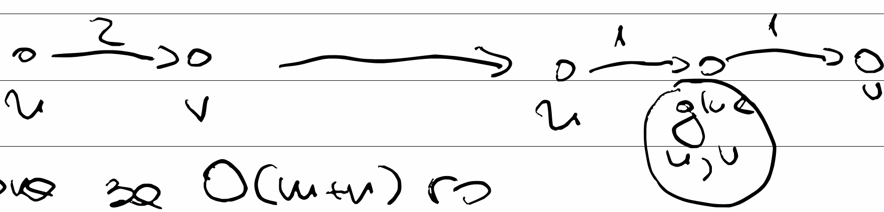
Dfs. Пускаме го. Графът е свързан, т.е. всички върхове и ребра ще обхождани. Като се връщаме от рекурсията по вече сивите върхове, то в обратната посока обхождаме ребрата т.е. в неор. свърз. граф имаме $n-1 \le m \le \frac{n.(n-1)}{2}$ и това е $2*m$.

$O(n+m)$ за dfs. Ще си записваме пътя.

4) Търси се най-къс път в граф с тегла на ребрата само 1 и 2.

<u>вар1:</u> Директно Dijkstra с Fibonacci heap $\to O(m+n\log n)$

<u>вар2:</u> Разцепваме всяко ребро с тегло 2 на 2 нови ребра и добавяме нов връх като ребрата са с тегла 1:



Това за $O(m+n)$ го правим така (обхождаме всяко ребро) и след това BFS за $O(m+n) \to$ тотал $O(m+n)$.

(5) Даден е граф $G=\langle V,E\rangle$, на който част от ребрата са ориентирани, а други не. Знаем, че ориентираните не образуват цикъл. Можем ли и как да ориентираме неориентираните ребра, така че да няма цикли?

Първо строим DAG като от оригиналният граф $G=\langle V,E\rangle$ оставяме само ориентираните:

$G' = \langle V, E' \rangle \to$ само ориентирани.

След това и правим topsort на графа G'.

$l = [v_1, v_2, \dots, v_n]$

$v_1 \to v_2 \to \dots \; v_n$

Можем останалите ребра да ги ориентираме по посоката на топо наредбата.

И трите стъпки са със сложност $\Theta(n+m)$

$\Theta(n+m)$.