

**АЛГОРИТМИ ВЪРХУ ГРАФИ**  
**ПРИМЕРНО КОНТРОЛНО № 3 ПО “ДИЗАЙН И АНАЛИЗ НА АЛГОРИТМИ” —**  
**ЗА СТУДЕНТИТЕ ОТ СПЕЦИАЛНОСТ “КОМПЮТЪРНИ НАУКИ”, 1. ПОТОК,**  
**СУ, ФМИ, ЛЕТЕН СЕМЕСТЪР НА 2019 / 2020 УЧЕБНА ГОДИНА**

Моделирайте входните данни чрез граф. Уточнете вида му — ориентиран, неориентиран; тегловен, нетегловен. Обяснете какво представляват върховете и ребрата на графа, а също и посоките и теглата на ребрата (ако има посоки и тегла). Формулирайте това, което се търси, в термините на теорията на графите и изберете подходящ алгоритъм от изучените на лекции. Ако алгоритъмът обхожда графа в ширина или в дълбочина, уточнете вида на обхождането. Всички алгоритми и алгоритмични схеми да бъдат назовани с българските си наименования.

**Задача 1.** Родителите на малкия Иванчо са му възложили няколко задачи, които той трябва да изпълни днес, и са го засипали с указания: “Преди да напазаруваш, нахрани котката.” “Преди да играеш, си напиши домашните.” “Преди да нахраниш котката, ѝ стопли храната.”

Помогнете на малкия Иванчо да реши в какъв ред да свърши всичко. За целта съставете алгоритъм с линейна времева сложност. **( 4 точки )**

**Задача 2.** На международна конференция има множество учени от различни държави. Всеки от участниците в конференцията знае по няколко езика.

а) Представете тази информация (кой участник какви езици говори) с помощта на граф, който съдържа възможно най-малко ребра. Тоест търси се възможно най-икономичен начин за задаване на входните данни (дължината на описанието на граф зависи най-вече от броя на ребрата, а не толкова от броя на върховете). Подкрепете преценката си за оптималност с изчисления за броя на ребрата, като сравните избраното от Вас представяне с някое друго. Вземете предвид, че хората са много повече от езиците; например на конференцията може да е имало 1000 участници, но те вероятно са говорили общо около 20 езика. **( 4 точки )**

б) Предложете алгоритъм с линейна времева сложност, който да проверява коректността на входните данни (тоест дали графът има избрания от Вас вид). **( 4 точки )**

в) Въпреки че всеки участник в конференцията знае по няколко езика, може да се случи някои двама участници да нямат общ език. В такъв случай те ще имат нужда от един или повече преводачи, така че всяка фраза да стигне от единия до другия участник, след като бъде преведена на един или няколко други езика. Понеже при всеки превод се губи нещо от смисъла на изречението (няма абсолютно точен превод), двамата участници без общ език искат да намерят редица от най-малък брой преводачи. Помогнете им, като съставите алгоритъм с линейна времева сложност за откриване на редицата. **( 4 точки )**

г) Да предположим допълнително, че за всяка двойка {участник, език} знаем число, показващо колко време е нужно на този участник, за да разбере или да формулира изречение на съответния език. Предложете възможно най-бърз алгоритъм, намиращ редица от преводачи, която ще осъществи с най-малко забавяне размяната на съобщения между двама избрани участници без общ език. **( 4 точки )**

## РЕШЕНИЯ

**Задача 1.** Входните данни се представят чрез ориентиран граф. Върхове са задачите, които малкият Иванчо трябва да свърши днес — да си напише домашните, да напазарува, да играе, да нахрани котката и т.н. Ребрата на графа съответстват на указанията на родителите: ако задачата  $x$  трябва да бъде свършена преди задачата  $y$ , графът съдържа ребро от  $x$  към  $y$ . Търсим подредба на върховете, при която всички ребра сочат надясно. Постигаме това за линейно време чрез *топологично сортиране*. То използва *обхождане в дълбочина*.

### Задача 2.

а) Информацията за това, кой участник какви езици говори, се представя най-естествено чрез *двуделен граф*: върховете в единия дял съответстват на участниците в конференцията, а върховете в другия дял — на езиците, говорени от тях. Графът съдържа ребро между върховете  $x$  и  $y$  точно тогава, когато човекът  $x$  говори езика  $y$ . Ребрата не са ориентирани.

Възможно е и друго представяне, но то не е така икономично: неориентиран граф, чиито върхове съответстват на хората, а ребрата свързват участниците с поне един общ език. Всяко ребро съдържа непразно множество от езици — общите езици на двамата участници, съответстващи на краищата на реброто.

Първото представяне съдържа по-малко ребра при типични стойности на параметрите. Ако на конференцията има например 1000 участници, които говорят общо 20 езика, тогава графът от първия вид ще съдържа най-много 20000 ребра. Този брой се достига само ако всеки участник говори всички езици. От друга страна, графът от втория вид ще съдържа най-малко ребра, ако всеки участник говори само един език и всички езици се говорят от равен брой участници. В този случай всеки език се говори от  $1000 : 20 = 50$  участници, всяка групичка от 50 участници съответства на клика с  $50 \cdot 49 : 2 = 1225$  ребра, а целият граф съдържа  $20 \cdot 1225 = 24500$  ребра — най-малкият възможен брой ребра за граф от втория вид. Тъй като  $24500 > 20000$ , то ясно е, че първото представяне е по-икономично.

б) Ако информацията е представена по първия начин (с помощта на двуделен граф), проверката за коректност на входните данни се свежда до *проверка за двуделност* на графа, тоест до оцветяване на върховете на графа в два цвята — цвят “човек” и цвят “език”. Можем да обходим графа било в ширина, било в дълбочина.

Ако информацията е представена по втория начин, трябва да проверим, че за всеки език ребрата, съдържащи този език, образуват пълен граф. Това може да стане с едно обхождане на графа (в ширина или в дълбочина), при което поддържаме по два брояча за всеки език: единият брой хората, говорещи съответния език, а другият — ребрата, съдържащи този език. Накрая сравняваме стойностите на двата брояча: ако  $k$  е броят на хората, а  $h$  — на ребрата,

трябва да е изпълнено равенството 
$$h = \frac{k(k-1)}{2}.$$

Това равенство е не само необходимо, но и достатъчно. Причината е, че дясната страна съдържа най-голямата възможна стойност на  $h$ : от всички неориентирани графи с  $k$  върха най-много ребра съдържа пълният граф.

в) Търсената редица представлява най-къс път (по брой ребра) между два дадени върха. Такъв път се намира с помощта на *търсене в ширина*. Ако използваме двуделен граф, броят на преводачите е равен на  $L / 2 - 1$ , където  $L$  е дължината на намерения най-къс път.

г) Времето за разбиране или формулиране на изречение е тегло на съответното ребро. Пак търсим най-къс път, но сега графът е тегловен и теглата на ребрата са положителни. Подходящ е *алгоритъмът на Дейкстра*.