V tomto článku jsem postupně ukázal aplikace běžné středoškolské matematiky ve webové grafice. Aktuální středoškolské kurikulum však není vzdělávání na středních školách v oblasti matematiky dostatečně zaměřeno na matice, které jsou v informatice klíčovým prvkem. Pokud by žáci zvládli matice, umožnilo by jim to snadněji pracovat s grafickými prvky v SVG formátu. Kromě webové grafiky se matice využívají třeba i v teorii grafů, nebo elektrotechnice.

Žáci všeobecných gymnázií jsou s maticemi seznamovány okrajově ve vyšších ročnících ve volitelných předmětech. Na jejich aplikace, ať už ve webové grafice, nebo teorii grafů již pak nezbývá čas. Domnívám se však, že na odborných školách, zaměřených na informační technologie či elektrotechniku by žáci měli být s maticemi seznámeni již na počátku studia. Navazující odborné předměty pak na těchto znalostech mohou dále stavět.

Literatura

[1] *Pomykalová, E.*: Matematika pro gymnázia: planimetrie. Učebnice pro střední školy, 4. upr. vyd., Prometheus, Praha, 2000.

[2] *Boček, L., Zhouf, J.*: Planimetrie. 2. rozš. vyd., Pedagogická fakulta, Univerzita Karlova, Praha, 2012.

[3] *Hejný, M., Jirotková, D., Vondrová, N.*: Geometrické transformace: (metoda analytická). Univerzita Karlova, Praha, 1997.

Rozsvícené lampy
(Úlohy z MO kategorie P, 46. část)

PAVEL TÖPFER

Matematicko-fyzikální fakulta UK, Praha

Pro dnešní pokračování dlouhodobého seriálu věnovaného vybraným problémům z Matematické olympiády kategorie P (programování) jsme zvolili jednu starší praktickou soutěžní úlohu. Jedná se o úlohu z celostátního kola 45. ročníku MO (školní rok 1995/96), tedy ze soutěže konané před téměř třiceti lety. Začneme jako obvykle původním zadáním úlohy, v němž jsme pro potřeby tohoto článku pouze zjednodušili tvar vstupních a výstupních dat.

* * * * * * * * * * * *

Veřejná prostranství ve městě jsou osvětlena N pouličními lampami. Každá z lamp má jednoznačně přiřazeno číslo od 1 do N. K zapínání a vypínání veřejného osvětlení slouží v řídicím středisku M přepínačů. Každý z přepínačů přepne najednou jednu nebo několik lamp. Přepnout lampu znamená zapnout ji, pokud zrovna nesvítí, a vypnout ji, jestliže momentálně svítí. Přepínač číslo i přepíná všechny lampy označené čísly od a_i do b_i, tzn. lampy s čísly ležícími v uvedeném uzavřeném intervalu.

Napište program, který zjistí, zda je možné pomocí přepínačů v řídicím středisku rozsvítit všechny lampy ve městě, jsou-li na začátku všechny tyto lampy zhasnuté.

Na prvním řádku vstupu jsou uvedena čísla N a M, kde N je počet lamp ve městě a M je počet přepínačů v řídicím středisku. Dalších M řádků vstupu obsahuje pro jednotlivé přepínače vždy dvojici čísel a_i, b_i (tedy pro každý přepínač je zadán interval čísel lamp, které se pomocí něho přepnou).

Výstupem programu je jeden řádek s jednou z následujících zpráv:

`Lze` – pokud je možné rozsvítit všechny lampy pomocí přepínačů v řídicím středisku,

`Nelze` – jestliže to není možné.

Příklady

Vstup 1:	Výstup 1:
5 3 1 3 2 5 2 3	`Lze`

Vstup 2:	Výstup 2:
10 5 1 9 2 10 3 10 4 10 5 10	`Nelze`

Primitivní řešení úlohy je založeno na prostém vyzkoušení všech možností. Postupně budeme procházet všechny možné skupiny přepínačů a pro každou z nich ověříme, zda přepnutím všech přepínačů z této skupiny rozsvítíme všechny lampy ve městě.

Stav všech lamp si budeme pamatovat v poli L, kde L[i] udává, zda je v pořadí i-tá lampa zapnutá, nebo ne. Pokud i-tá lampa svítí, prvek L[i] bude mít hodnotu `True`, v opačném případě `False`. Na začátku výpočtu budou všechny hodnoty v poli L nastaveny na `False`. Při zkoumání zvolené skupiny přepínačů stačí pro každý přepínač j z této skupiny změnit všechny hodnoty v poli L na indexech od a_j do b_j na opačné. Tím vlastně přímo modelujeme průběh přepínaní lamp při použití vybraných přepínačů. Nemusíme se nijak zabývat tím, v jakém pořadí přepínače použijeme, protože jejich pořadí nemá žádný vliv na výsledek.

Řešením úlohy budeme rozumět takovou skupinu přepínačů, že pokud je všechny přepneme, všechny lampy budou nakonec svítit (tzn. všechny prvky v poli L budou mít hodnotu `True`). Jakmile během postupného zkoušení jednotlivých skupin přepínačů najdeme řešení úlohy, program odpoví `Lze` a výpočet ukončíme. Pokud vyzkoušíme všechny možnosti, ale řešení nenajdeme, program odpoví `Nelze`.

Popsaný postup je sice velmi jednoduchý, ale je také velmi neefektivní. Počet všech různých podmnožin M-prvkové množiny je totiž 2^M. V nejhorším případě musíme vyzkoušet všech 2^M skupin vybraných přepínačů, každá tato skupina bude obsahovat $\mathrm{O}(M)$ přepínačů a každý z nich přepíná $\mathrm{O}(N)$ lamp. Celkovou časovou složitost algoritmu můžeme proto odhadnout jako $\mathrm{O}(N \cdot M \cdot 2^M)$. Je tedy exponenciální vzhledem k počtu přepínačů. Takový algoritmus můžeme reálně použít pouze pro velmi malé hodnoty M.

Ukážeme si proto jiný postup řešení úlohy, který je podstatně rychlejší, ale vyžaduje trochu náročnější počáteční rozbor situace. Nejprve budeme řešit obecnější úlohu: předpokládejme, že na začátku nemusí být všechny lampy vypnuté. Když budeme mít odvozen algoritmus pro řešení úlohy s obecným počátečním stavem lamp, můžeme ho nakonec samozřejmě použít na počáteční situaci se všemi lampami zhasnutými, jako je tomu v naší soutěžní úloze.

Pro evidenci stavu jednotlivých lamp budeme používat stejné pole L, jaké jsme zavedli výše. Hodnoty `True`/`False` v poli L nám popisují obecnou počáteční situaci, pro kterou chceme nalézt řešení. Rozlišíme dva případy podle toho, v jakém stavu se nachází první lampa. Předpokládejme nejprve, že je vypnutá. Pokud žádný interval nezačíná lampou 1, úloha zcela jistě nemá řešení, protože první lampu vůbec nedokážeme rozsvítit. V opačném případě najdeme nejkratší interval, který začíná lampou 1, a jeho konec označíme b. Přepínač odpovídající tomuto intervalu použijeme (ale pozor, v definitivním řešení úlohy ho možná později ještě vyměníme za jiný!) a změníme tak všechny hodnoty $\mathrm{L}[i]$ z intervalu od 1 do b na opačné. Zároveň s tím všem ostatním intervalům začínajícím lampou 1 změníme jejich začátek na $b+1$. Tím dostaneme upravenou úlohu pro lampy $2, 3, \ldots, N$. Ukážeme, že tato upravená úloha má řešení právě tehdy, má-li řešení naše původní úloha.

Než přikročíme k důkazu uvedeného tvrzení, vysvětlíme si, jaký význam má poznámka, že právě použitý přepínač $\langle 1, b\rangle$ možná v budoucnu ještě vyměníme za jiný. Mohlo by se totiž stát, že v řešení upravené úlohy použijeme některý z přepínačů, kterým jsme právě zkrátili jejich interval z původního $\langle 1, c\rangle$ na $\langle b+1, c\rangle$. Takový přepínač ovšem v původní úloze neexistuje. Když jsme ale nejprve použili přepínač $\langle 1, b\rangle$ a nyní chceme použít přepínač $\langle b+1, c\rangle$, můžeme se stejným výsledkem místo těchto dvou přepínačů použít jeden přepínač $\langle 1, c\rangle$, který máme k dispozici. Ve výsledném řešení tedy nakonec původně uvažovaný přepínač $\langle 1, b\rangle$ nepoužijeme a nahradíme ho přepínačem $\langle 1, c\rangle$.

Nyní se již můžeme věnovat důkazu výše uvedeného tvrzení. Dokazované tvrzení je ekvivalence, budeme proto dokazovat zvlášť implikaci jedním směrem a zvlášť druhým směrem. Nejprve předpokládejme, že upravená úloha má řešení R. Pokud přepneme lampy z původní úlohy použitím přepínačů tvořících řešení R, budou svítit jen lampy s čísly většími než b. Lampa 1 byla původně vypnutá a vypnutá zůstane i nadále, protože žádný přepínač z řešení R ji neobsahuje. Lampy z intervalu od 2 do b měly v upravené úloze převrácené výchozí hodnoty oproti původní úloze, takže když budou v upravené úloze nakonec svítit, tak v původní úloze budou po použití přepínačů R zhasnuté. Předpokládejme, že řešení R obsahuje k intervalů, kterým jsme začátek změnili z 1 na $b+1$. Takové přepínače ovšem v původní úloze neexistují. Řešení S vznikne z R tak, že těmto k intervalům změníme jejich začátek zpět na 1. Jestliže je k liché, každou lampu z intervalu $\langle 1, b\rangle$ jsme tím oproti řešení R přepnuli ještě lichý počet krát, proto bude zapnutá. V tomto případě je tedy S správným řešením naší původní úlohy. Pokud je k sudé, lampy z intervalu $\langle 1, b\rangle$ zůstanou vypnuté a proto k řešení S je třeba ještě přidat přepínač ovládající interval lamp od 1 do b. V obou případech jsme z řešení R upravené úlohy sestrojili řešení S původní úlohy. Tím jsme dokázali, že má-li upravená úloha řešení, potom má řešení také původní úloha.

Předpokládejme teď naopak, že původní úloha má řešení S. Počet přepínačů z S, které obsahují lampu 1, je jistě lichý. Když všem jejich intervalům změníme začátek z 1 na $b+1$ (a vypustíme intervaly nulové délky), dostaneme řešení R upravené úlohy. Uvedeným zkrácením intervalů jsme totiž změnili hodnoty všech lamp od 2 do b, zatímco ostatních lamp se žádná změna nedotkla. Tím je dokázána druhá implikace a tedy i celé tvrzení.

Zbývá popsat řešení úlohy v případě, že je první lampa zapnutá. Protože lampa 1 svítí, není zapotřebí, aby se dala nějakým přepínačem přepnout. Jestliže neexistuje žádný přepínač, který ovládá lampu 1, úloha má řešení právě tehdy, má-li řešení úloha pro lampy $2, 3, \ldots, N$ a pro stejnou množinu přepínačů. Pokud existuje alespoň jeden interval začínající lampou 1, opět vezmeme nejkratší z takových intervalů $\langle 1, b\rangle$ a změníme začátky ostatních intervalů stejně jako v předchozím případě na hodnotu $b+1$. Tentokrát ale nezměníme hodnoty v poli L, protože přepínač odpovídající nejkratšímu intervalu $\langle 1, b\rangle$ nepoužijeme, svítící lampu 1 nechceme přepnout (ovšem může se stát podobně jako v předchozím případě, že toto svoje rozhodnutí ještě v budoucnu změníme). Takto upravená úloha má opět řešení právě tehdy, když má řešení původní úloha. Důkaz je obdobný jako v předchozím případě.

Nyní se můžeme vrátit k původnímu zadání soutěžní úlohy, v němž jsou na začátku všechny lampy zhasnuté. Naším úkolem není nalézt řešení, ale pouze zjistit, zda nějaké řešení existuje. Nastavíme proto všechny prvky pole L na `False` a podle výše uvedených úvah budeme postupovat od první lampy až k poslední, přičemž modifikujeme obsah pole L a začátky příslušných intervalů. Pokud se v některém kroku dostaneme do situace, že upravená úloha nemá řešení, pak nemá řešení ani původní zadaná úloha. Jestliže tato situace nikde nenastane, pak řešení úlohy existuje.

Popsaný algoritmus provádí postupně N kroků výpočtu pro jednotlivé lampy. V každém kroku musíme projít seznam všech M přepínačů a upravit ty z nich, které ovládají aktuálně zpracovávanou lampu. Zkrácení intervalu každého takového přepínače má konstantní časovou složitost, ale jeden přepínač s nejkratším intervalem vyžaduje projít a přepnout všechny jím ovládané lampy, kterých je $\mathrm{O}(N)$. Asymptotickou časovou složitost algoritmu proto můžeme odhadnout jako $\mathrm{O}(N \cdot (N+M))$. Prostorová složitost algoritmu je $\mathrm{O}(N+M)$, neboť pracujeme se seznamem L délky N (obsahuje aktuální stav jednotlivých lamp) a se seznamem informací o jednotlivých přepínačích délky M.

Ukázkový program v Pythonu je přímou implementací popsaného algoritmu.

```
n, m = [int(_) for _ in input().split()]
prep = []                              # intervaly přepínačů
for j in range(m):
    prep.append([int(_) for _ in input().split()])
L = [False] * (n+1)                    # příznak zda lampa svítí

ok = True                              # existence řešení
for i in range(1, n+1):                # aktuální lampa
    b = n+1   # konec nejkratšího intervalu se začátkem i
    for j in range(m):
        if prep[j][0] == i and prep[j][1] < b:
            b = prep[j][1]
    if b == n+1:               # žádný interval se začátkem i
        ok = L[i]
        if not ok: break
    else:
        if not L[i]:           # přepneme lampy od i do b
            for k in range(i, b+1):
                L[k] = not L[k]
        j = 0
        while j < m:  # změníme začátky intervalů i -> b+1
            if prep[j][0] == i:
                prep[j][0] = b+1
            if prep[j][0] > prep[j][1]: # prázdný interval
                prep[j] = prep[m-1]
                m -= 1
            else:
                j += 1

if ok:
    print("Lze")
else:
    print("Nelze")
```