

# MATEMATICKÁ OLYMPIÁDA NA STREDNÝCH ŠKOLÁCH

50. ROČNÍK, školský rok 2000/2001

Zadania úloh celoštátneho kola kategórie P

1. súťažný deň

Celoštátne kolo 50. ročníka MO kategórie P sa koná v dňoch 4. – 7. 4. 2001. Na riešenie úloh máte 4.5 hodiny čistého času. Riešenie každého príkladu musí obsahovať (pokiaľ nie je v zadaní uvedené ináč):

- **Popis riešenia**, to znamená slovný popis použitého algoritmu, argumenty zdôvodňujúce jeho správnosť (prípadne dôkaz správnosti algoritmu), diskusiu o efektívite vášho riešenia (časová a pamäťová zložitost'). Slovný popis riešenia musí byť jasný a zrozumiteľný i bez nahliadnutia do samotného zápisu algoritmu (do programu).
- **Program**. V úlohách **P-III-1** a **P-III-2** treba uviesť dostatočne podrobný zápis algoritmu, najlepšie v tvare zdrojového textu najdôležitejších častí programu v jazyku Pascal alebo C. Zo zápisu môžete vynechať jednoduché operácie ako vstupy, výstupy, implementáciu jednoduchých matematických vzťahov a pod. V úlohe **P-III-3** uveďte namiesto zdrojového textu zápis dlaždicového programu.

Hodnotí sa nielen správnosť programu, ale tiež kvalita popisu riešenia a efektivita zvoleného algoritmu.

## P-III-1

(Veže)

Pán Kašparov bol náruživý hráč šachu. Keďže mu často chýbal rovnocenný protivník a hrať sám proti sebe ho už nebavilo (vždy si odhalil všetky premyslené pasce), vymyslel si nasledujúcu hru: Na šachovnici s rozmermi  $N \times N$  treba rozmiestniť  $N$  veží tak, aby sa žiadne dve neohrozovali. Aby hra nebola až taká jednoduchá, je pre každú vežu určený obdĺžnik, do ktorého ju treba umiestniť.

Vašou úlohou je navrhnúť algoritmus, ktorý bude hrať túto hru. Na vstupe dostane rozmer šachovnice  $N$  (čo je zároveň aj počet veží) a ďalej popis  $N$  obdĺžnikov. Jeden obdĺžnik je popísaný štvoricou čísel  $A_x, A_y, B_x, B_y$ ,  $1 \leq A_x \leq B_x \leq N$ ,  $1 \leq A_y \leq B_y \leq N$ , kde  $A_x, A_y$  sú súradnice ľavého horného rohu obdĺžnika a  $B_x, B_y$  sú súradnice pravého dolného rohu. Riadky číslujeme od 1 do  $N$  zhora dole, stĺpce od 1 do  $N$  zľava doprava. Na výstup má algoritmus vypísať jedno ľubovoľné prípustné rozloženie veží alebo podať správu o tom, že vyhovujúce rozloženie neexistuje.

### Príklad 1:

#### Vstup:

N=4  
1 1 1 1  
4 4 4 4  
1 1 3 3  
3 2 4 4

#### Výstup:

Vyhovuje napr. nasledovné  
rozmiestnenie veží:  
(1, 1), (4, 4), (2, 2), (3, 3).

## Príklad 2:

### Vstup:

$N = 3$   
1 1 3 1  
2 1 3 1  
2 2 3 3

### Výstup:

Vyhovujúce rozmiestnenie  
veží neexistuje.

## P-III-2

(Nuly a jednotky)

Napište program, ktorý na vstupe dostane prirodzené číslo  $N$  a nájde najmenšie prirodzené číslo  $x$  také, že  $x$  je deliteľné číslom  $N$  a zároveň zápis čísla  $N$  v desiatkovej sústave obsahuje iba cifry nula a jedna. V prípade, že také číslo neexistuje, váš program by o tom mal podať vhodnú správu.

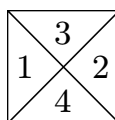
Napríklad pre  $N = 6$  je hľadaným číslom  $x$  číslo 1110.

## P-III-3

(Dlaždice)

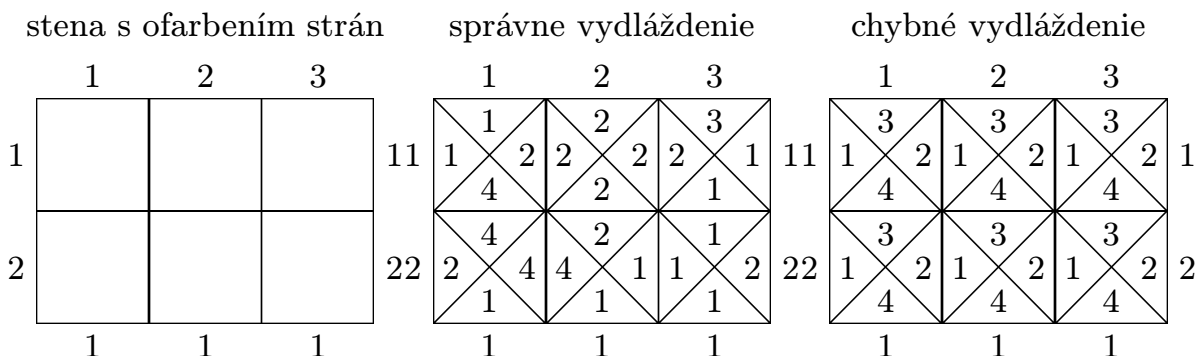
(Definícia dlaždicových programov je rovnaká ako v krajskom kole, iba je v nej navyše vysvetlené, ako sa dajú dlaždicové programy používať na výpočet funkcie [posledný odsek pred príkladom] a uvádza sa tam príklad takéhoto použitia.)

Najprv niekoľko definícií: *Dlaždice* sú rovnako veľké štvorce s ofarbenými hranami. Konkrétne priradeniu farieb hranám dlaždice budeme hovoriť *typ* dlaždice a budeme ho zapisovať ako usporiadanú štvoricu  $(l, p, h, d)$  udávajúcu farby v poradí ľavej, pravej, hornej a dolnej hrany. Aby sme si uľahčili prácu, budeme farby označovať rôznymi symbolmi — písmenami, číslami a podobne. Dlaždica typu  $(1, 2, 3, 4)$  bude teda vyzeráť nasledovne:



Priestor, ktorý budeme dlaždiť (budeme mu hovoriť *stena*), má tvar obdĺžnika o veľkosti  $m \times n$  ( $m$  aj  $n$  sú prirodzené čísla; jednotkou dĺžky nech je dĺžka hrany dlaždice). Strany obdĺžnika sú rozdelené na úseky jednotkovej dĺžky a každému úseku je opäť priradená farba. Naším cieľom je *vydlaždiť* stenu dlaždicami tak, aby v každom z  $m \cdot n$  jednotkových štvorcov steny bola umiestená práve jedna dlaždica, susedné dlaždice se dotýkali vždy hranami tej istej farby a rovnako krajné dlaždice priliehali k okraju steny vždy hranou takej farby, akú má aj príslušný úsek okraja steny. Dlaždice nie je povolené otáčať.

### Príklad:



Pomocou dláždenia môžeme ľahko riešiť úlohy, ktorých výsledkom je odpoveď *áno* alebo *nie*: zostavíme vhodnú množinu typov dlaždíc (tá je pre daný problém pevná – nezávisí na vstupe), vezmeme vhodne veľkú stenu, jej horný okraj ofarbíme podľa vstupu nášho problému, ostatné okraje ponecháme jednofarebné a budeme sa pýtať, či je túto stenu možné vydláždiť alebo nie. Pritom chceme, aby tento výsledok bol zhodný s riešením našej úlohy.

Aby sme sa nemuseli zaoberať tým, ako presne veľkú stenu máme zvoliť pre ten či onen vstup úlohy, budeme šírku steny voliť vždy rovnakú, ako je dĺžka vstupu (horný okraj teda bude celý zaplnený vstupom), zatiaľ čo výšku steny použijeme najmenšiu, pre ktorú existuje vydláždenie s použitím našej sady dlaždíc.

Keď tento spôsob počítania porovnáme s klasickým programovaním, zistíme, že zvolená sada dlaždíc tvorí v našom modeli niečo podobné programu a potrebná výška steny vzdialene odpovedá dobe behu výpočtu – budeme sa preto snažiť, aby v našich riešeniach bola čo najmenšia.

Formálne povedané, *dlaždicový program* je usporiadaná štvorica

$$D = (T, l_0, p_0, d_0),$$

kde  $T$  je konečná množina typov dlaždíc  $\{(l_1, p_1, h_1, d_1), \dots, (l_k, p_k, h_k, d_k)\}$  a  $l_0$ ,  $p_0$  a  $d_0$  sú okrajové farby. *Rozhodovacou úlohou*  $P(x)$  rozumieme úlohu zistiť, či vstup  $x$  (konečná postupnosť symbolov, resp. farieb z vopred určenej konečnej množiny) má požadovanú vlastnosť  $P$ . Hovoríme, že dlaždicový program *rieši rozhodovaciu úlohu*  $P(x)$ , ak platí, že  $P(x) = \text{áno}$  práve vtedy, keď existuje  $v > 0$  také, že je možné vydláždiť dlaždicami typov obsiahnutých v množine  $T$  stenu veľkosti  $|x| \times v$  s hornou hranou ofarbenou vstupom  $x$  a ľavou, pravou a dolnou hranou ofarbenou postupne farbami  $l_0$ ,  $p_0$  a  $d_0$ . Z každého typu je možné použiť ľubovoľne mnoho dlaždíc. *Zložitosťou* programu  $D$  pre daný vstup  $x$  nazveme najmenšie  $v$ , pre aké to je možné; pokiaľ také  $v$  neexistuje, a teda  $P(x) = \text{nie}$ , definujeme zložitost ako nulovú. Zložitost programu je funkcia dĺžky vstupu  $n$ , ktorej hodnota udáva maximum zo zložitostí programov pre jednotlivé vstupy tejto dĺžky.

Dlaždicové programy sa dajú používať nielen na riešenie rozhodovacích problémov, ale aj na výpočet hodnôt funkcií. Výpočet funkcie  $f(x)$  totiž môžeme ľahko previesť na rozhodovací problém  $P(x, y) = \text{“je } y = f(x)\text{?”}$ , o ktorom budeme vedieť, že pre každé  $x$  bude  $P(x, y)$  platiť práve pre jednu hodnotu  $y$ . Navyše môžeme dlaždicovému programu  $x$  a  $y$  zadať ako jeden vstup tak, že farby dlaždíc nebudú zodpovedať hodnotám, ale ich usporiadaným dvojiciam.

*Príklad:*

Zostrojme dlaždicový program, ktorý pre dané číslo zapísané v dvojkovej sústave zráta dvojkový zápis tohto čísla vydeleného tromi (predpokladajme, že je deliteľné 3 bez zvyšku). Inými slovami máme o danej postupnosti dvojíc  $(x_1, y_1), \dots, (x_n, y_n)$  zistiť, či  $\langle y_1, \dots, y_n \rangle = \langle x_1, \dots, x_n \rangle / 3$ .

Riešenie založíme na klasickom algoritme na písomné delenie (ten nezávisí od použitej číselnej sústavy): zvolíme  $z_0 = 0$  a postupne pre všetky  $k$  budeme rátať hodnoty  $z_k = (2 \cdot z_{k-1} + x_k) \bmod 3$  a  $y_k = \lfloor (2 \cdot z_{k-1} + x_k) / 3 \rfloor$ . Teraz dokážeme indukciou, že pre každé  $k$  je

$$\langle x_1, \dots, x_k \rangle = 3 \cdot \langle y_1, \dots, y_k \rangle + z_k.$$

Pre  $k = 0$  rovnosť platí. Ak platí pre  $k - 1$ , potom pre  $k$  dostávame:

$$\begin{aligned}\langle x_1, \dots, x_k \rangle &= 2 \cdot \langle x_1, \dots, x_{k-1} \rangle + x_k = \\ &= 2 \cdot (3 \cdot \langle y_1, \dots, y_{k-1} \rangle + z_{k-1}) + x_k = \\ &= 3 \cdot 2 \cdot \langle y_1, \dots, y_{k-1} \rangle + 2 \cdot z_{k-1} + x_k = \\ &= 3 \cdot 2 \cdot \langle y_1, \dots, y_{k-1} \rangle + 3 \cdot y_k + z_k = \\ &= 3 \cdot \langle y_1, \dots, y_k \rangle + z_k.\end{aligned}$$

Teraz stačí zvoliť si nasledujúcu sadu dlaždíc:

$$T = \left\{ \begin{array}{|c|c|} \hline \begin{array}{c} xy \\ a \quad b \\ \bullet \end{array} & ; x, y \in \{0, 1\}, a \in \{0, 1, 2\}, b = (2a + x) \bmod 3, y = \lfloor (2a + x)/3 \rfloor \\ \hline \end{array} \right\},$$

Ľavý a pravý okraj budú mať farbu 0, spodný farbu  $\bullet$ .

Z takýchto dlaždíc sa dajú zložiť len jednoriadkové steny, lebo  $\bullet$  nie je na hornom okraji žiadnej dlaždice. V prípustnom vydláždení má  $k$ -ta dlaždica na svojom pravom okraji  $z_k$  a pre dvojicu  $(x_k, y_k)$  na jej hornom okraji platí:  $y_k = \lfloor (2 \cdot z_{k-1} + x_k)/3 \rfloor$ . Inými slovami toto vydláždenie zodpovedá presne hodnotám vypočítaným našim algoritmom, teda aj požadovanému výsledku. Tým je problém vyriešený.

### Súťažná úloha:

Zostrojte dlaždicový program, ktorý bude usporadúvať postupnosť núl a jednotiek vzostupne, to znamená, že na postupnosť dvojíc núl a jednotiek  $(x_1, y_1), \dots, (x_n, y_n)$  odpovie *áno* práve vtedy, keď  $y_1, \dots, y_n$  je postupnosť, ktorú dostaneme vzostupným usporiadaním postupnosti  $x_1, \dots, x_n$ , teda platí  $y_1 \leq \dots \leq y_n$  a postupnosti  $x$  a  $y$  obsahujú tie isté prvky, nanajvýš v inom poradí.

### Príklad:

Na postupnosť  $(1, 0), (0, 0), (0, 0), (1, 1), (0, 1), (1, 1)$  program odpovie *áno*, na  $(1, 1), (0, 0)$  *nie*, na  $(1, 0), (1, 1)$  taktiež *nie*.

## SLOVENSKÁ KOMISIA MATEMATICKEJ OLYMPIÁDY

### 50. ROČNÍK MATEMATICKEJ OLYMPIÁDY

#### Zadania celoštátneho kola kategórie P

Vydala IUVENTA – zariadenie pre voľný čas detí, mládeže i dospelých MŠ SR  
pre vnútornú potrebu Ministerstva školstva SR  
Programom T<sub>E</sub>X sadzbu pripravil Michal Forišek

Autori príkladov: Jan Kára  
Daniel Král  
Martin Mareš

© Slovenská komisia matematickej olympiády, 2001