

Korešpondenčný seminár z programovania XIX. ročník, 2001/2002

Katedra vyučovania informatiky FMFI UK,
Mlynská Dolina, 842 48 Bratislava

*KSP finančne podporuje nadácia Open Society Fund Bratislava
IUVENTA – zariadenie pre voľný čas detí, mládeže a dospelých
MICROSTEP spol. s.r.o.*

Propozície

Korešpondenčný seminár z programovania (KSP) je súťaž programátorov – stredoškôľakov. Jej cieľom je zdokonaľiť žiakov, ktorí už vedia programovať alebo sa práve programovať učia, v programovaní a v algoritmickom myslení.

Kategórie: Korešpondenčný seminár z programovania pozostáva z dvoch kategórií – **KSP a KSP-Z**. Kategória KSP-Z je určená pre začínajúcich riešiteľov. V tomto ročníku sa jej môže zúčastniť každý stredoškôľák, ktorý v žiadnom z predchádzajúcich ročníkov v kategórii KSP nezískal za celý ročník viac ako **30 bodov**, v kategórii KSP-Z viac ako **90 bodov**, a nezúčastnil sa sústredenia KSP. Kategórie KSP sa môžu zúčastniť všetci stredoškôľáci bez obmedzenia (aj riešitelia KSP-Z). Obzvlášť radi privítame účasť dievčat.

Sústredenia: Najlepších riešiteľov oboch kategórií pozývame každoročne na dve týždenné sústredenia (pokiaľ sa ich podarí zorganizovať). Na jesenné sústredenie nepozývame riešiteľov, ktorí už ukončili strednú školu. Pokiaľ viete o niekom, kto by mohol tieto sústredenia čo i len čiastočne sponzorovať, dajte nám o ňom (a prípadne jemu o nás) vedieť.

Organizácia súťaže: Kategória KSP aj kategória KSP-Z sú organizované v štyroch kolách, ktoré sú rozdelené na **dva samostatné polroky po dvoch kolách**. Na jarné sústredenie teda budú pozývaní najlepší riešitelia podľa počtu bodov za 1. a 2. kolo zimnej časti a na jesenné sústredenie podľa počtu bodov za 1. a 2. kolo letnej časti. Každé kolo obsahuje päť úloh. Príklady každého kola (nie nutne všetky) je treba vyriešiť a do určeného termínu poslať na našu adresu. Riešenia odoslané po tomto termíne nemusia byť hodnotené, prípadne môžu byť riešiteľovi strhnuté body za neskoré odoslanie riešenia. Riešiteľom, ktorí sa zúčastnili niektorého kola súťaže, pošleme zadania ďalšieho kola, výsledkovú listinu a komentáre k riešeniam. Ďalšieho kola sa však môžu zúčastniť aj tí žiaci, ktorí sa predchádzajúceho kola nezúčastnili.

Požiadavky na riešenia: Základom riešenia každého príkladu (ak nie je v zadaní uvedené inak) je **listing programu** v ľubovoľnom vyššom programovacom jazyku (najlepšie Pascal, C/C++). Na programe nás zaujíma najmä jeho korektnosť (dáva program pre každý vstup správny výsledok?) a efektivita (koľko času a pamäte potrebuje na spracovanie vstupu určitej veľkosti). Dôraz kladte na algoritmickú časť programu, pri načítavaní vstupu a vypisovaní výstup nemusíte úplne presne dodržiavať formát zo zadania. Rovnako dôležitou súčasťou riešenia je **slovný popis** riešenia. Slovný popis by mal obsahovať popis použitého algoritmu, prípadne popis kľúčových dátových štruktúr. Mal by byť natoľko jasný a zrozumiteľný, aby bolo podľa neho možné napísať program rovnako efektívny, ako ten váš. Ďalej vyžadujeme **zdôvodnenie** (dôkaz) **správnosti** použitého algoritmu a odhad časovej a pamätevej zložitosti algoritmu (t.j. koľko času a pamäte potrebuje váš program v závislosti od veľkosti vstupných dát).

Formálna úprava: **Diskety nám neposielajte.** Svoje riešenia píšete na papieri formátu A4. Prvý list každého riešenia má obsahovať hlavičku, v ktorej je meno riešiteľa, číslo príkladu, kategória, počet listov príkladu a škola. Ďalšie listy obsahujú meno, číslo príkladu a číslo listu. Ak má príklad niekoľko listov, zopnite ich spolu (v ľavom hornom rohu), aby sa nestratili.

Nezopínajte listy patriace rôznym príkladom a už vôbec nepíšte riešenia viacerých príkladov na ten istý list papiera.

Bodovanie: Maximálne počty bodov pre jednotlivé príklady sa v zadaniach neuvádzajú, pohybujú sa od 10 do 25 bodov podľa náročnosti príkladu. KSP je súťaž jednotlivcov, preto ak sa vyskytnú „kolektívne riešenia“, bude za ne udelených patrične menej bodov.

Evidenčný lístok: Každý súťažiaci musí poslať spolu s riešeniami prvého kola vyplnený evidenčný lístok. Veľmi dôležitou informáciou na tomto lístku je adresa pre korešpondenciu. Na túto adresu budeme posielat opravené riešenia a zadania ďalších kôl. Ak táto adresa je iná, ako vaša domáca adresa (ak bývate napr. na internáte), napíšte nám aj svoju domácu adresu, aby sme vám mohli písať aj cez prázdniny. Nezabudnite tiež uviesť ročník štúdia a označiť kategórie, ktorých sa chcete zúčastniť. Tí z vás, ktorí máte prístup k Internetu, môžete evidenčný lístok vyplniť na adrese <http://www.ksp.sk/ksp/prihlaska.php>.

Poštovné náklady: Žiaľ nie je v našich silách hradiť všetky výdavky na poštovné (rozposielanie zadání, riešení a vzorových riešení). Preto sme zaviedli „účastnícky poplatok“. Účastnícky poplatok je **15,- Sk** za každú sériu, ktorej sa riešiteľ zúčastní a posiela sa spolu s **riešeniami úloh formou poštových známok v rozumných hodnotách, napr. 2,- a 5,- Sk**. Účastníkom, ktorí nám nepošlú známky, hrozí, že od nás nedostanú žiadnu poštu.

Tešíme sa na vaše riešenia a prajeme vám veľa dobrých nápadov

KSPáci

Adresa: KSP

Katedra vyučovania informatiky FMFI UK

Mlynská dolina

842 48 Bratislava 4

E-mail: ksp@ksp.sk

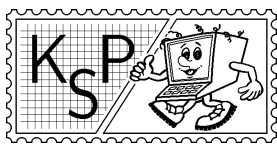
URL: <http://www.ksp.sk/ksp>

Telefón: 02 / 60 29 52 10

Mailing list eKSPres

Mailing list eKSPres je určený riešiteľom a opravovateľom KSP, ako aj najrôznejším priaznivcom seminára. Dá sa na ňom dozvedieť o rôznych akciách organizovaných vedúcimi KSP (napríklad splav), diskutovať o zaujímavých problémoch (s výnimkou príkladov aktuálnej série KSP!) a pod.

- Ak sa chcete doňho prihlásiť, musíte mať e-mailovú adresu. Z nej pošlite na adresu eKSPres-request@ksp.sk mail obsahujúci jediné slovo **SUBSCRIBE** v tele správy. Od tohoto okamihu vám budú na vašu adresu chodiť všetky správy z mailing listu eKSPres.
- Ak chcete ostatným poslať mail, pošlite mail na adresu eKSPres@ksp.sk. Váš mail príde všetkým členom eKSPres-u, preto sa pred jeho odoslaním najprv zamyslite nad jeho vhodnosťou.



Korešpondenčný seminár z programovania
XIX. ročník, 2001/2002
Katedra vyučovania informatiky FMFI UK,
Mlynská Dolina, 842 48 Bratislava

KSP finančne podporuje nadácia Open Society Fund Bratislava
IUVENTA – zariadenie pre voľný čas detí, mládeže a dospelých
MICROSTEP spol. s.r.o.

Kategória Z

Príklady 1. kola letnej časti

Milé naše riešiteľky, milí naši riešitelia,
dostávajú sa vám do rúk zadania 1. kola letnej časti 19. ročníka KSP-Z. Prečítajte si
pozorne propozície súťaže a hor sa riešiť. Svoje riešenia nám posielajte najneskôr do **25.**
marca. Nezabudnite k nim priložiť známky v hodnote 15,- Sk.

Páchnuci slimák sa nesmie dotknúť Harryho Pottera!

KSPáci

1. Zanzibarské jednosmerky

Zanzibar je veľké a prastaré mesto. Začali ho stavať ešte pred našim letopočtom. Začiatkom letopočtu však naši prapredkovia nemali ani kúsok citu a pochopenia pre modernú urbanizáciu, automobilovú dopravu, potreby dnešného moderného človeka a vôbec. Zaoberali sa hlavne tým, ako na čo najmenšiu plochu nadžgať čo najviac chatrčí. Žiaľ, týmto vývojom sa stalo, že uličky v Zanzibare sú priveľmi úzke pre obojsmernú automobilovú dopravu.

Začiatkom 21. storočia sa problém vyostřil natoľko, že mestská rada bola nútená narýchlo zaviesť v meste systém jednosmeriek. Zbastlená reforma však, zdá sa, všetko iba zhoršila. Zapeklitým problémom totiž ostáva, či sa v meste dá dostať autom z ľubovoľného miesta na ľubovoľné iné tak, aby sa dodržal predpísaný smer jazdy. Zistite, či je to skutočne tak!

Úloha: Zanzibar si možno predstaviť ako systém jednosmeriek, ktoré vedú medzi križovatkami. Zadané je N – počet križovatiek a M – počet jednosmeriek. Ďalej je daných M jednosmeriek. Jednosmerka je daná ako dvojica čísel križovatiek x, y ($1 \leq x, y \leq N$), medzi ktorými vedie. Jazdiť sa smie po tejto ulici len v smere od križovatky x ku križovatke y . Napíšte program, ktorý zistí, či sa možno medzi dvoma ľubovoľnými križovatkami v meste dostať autom tak, aby sme dodržali dopravné predpisy.

Príklad:

Vstup:

$N = 6, M = 7$

1 2

2 3

3 1

4 5

5 6

6 4

1 4

Výstup:

Nie, nedá sa dostať medzi ľubovoľnými dvoma križovatkami.

(Pozn.: napríklad sa nedá dostať z križovatky 5 na križovátku 3)

2. Zábudliví trpaslíci

„A čo si nám to chcel povedať?“, pýtali sa (už trochu nahnevaní, pretože sa to pýtajú už siedmy krát) trpaslíci chudáka Zedrika. No nestávalo sa to len jemu, ale viacerým trpaslíkom

– niekedy, keď si dohodli stretnutie pod jedným zo stromčekov, si zrazu nemohli podaktorí spomenúť, čo chceli ostatným povedať. Jednoducho – trpasličia skleróza. Nebola to choroba nebezpečná, ale tiež nie veľmi príjemná. Každý trpaslík mal doma na chladničke lístok, kam si písal všetky dôležité veci, aby na ne nezabudol. No keď išiel na stretnutie pod stromček, chladničku si so sebou pochopiteľne nebral, a potom sa mohlo stať, že kým prišiel k stromčeku, zabudol, čo vlastne chcel.

Preto sa rada starších rozhodla, že s tým treba niečo spraviť. Ideálne zjavne bude, ak sa budú stretávať pod takým stromčekom, aby všetci trpaslíci spolu prešli čo najmenšiu vzdialenosť.¹ Trpasličia dedinka má len jednu ulicu, na ktorej sú domčeky trpaslíkov. Medzi každými dvoma domčekmi rastie stromček, pod ktorým sa dá stretávať. Polohy domčekov, ako aj stromčekov, sú celé čísla, ktoré určujú vzdialenosť daného objektu od začiatku ulice.

Úloha: Pomôžte rade starších a napíšte im program, ktorý im povie, kde sa majú trpaslíci stretnúť, aby toho spolu čo najmenej prešli (a čo najmenej zabudli). Vstupom programu sú utriedené pozície N domčekov, z ktorých z každého pôjde na miesto stretnutia jeden trpaslík, a ďalej utriedené pozície $N - 1$ stromčekov – jeden medzi susednými dvoma domčekmi. Program má za úlohu zistiť, pri ktorom zo stromčekov sa majú trpaslíci stretnúť.

Príklad:

Vstup:

$N = 5$

domčeky: 0, 3, 10, 12, 16

stromčeky: 1, 6, 11, 13

Výstup:

Najmenej sa nachodia, ak sa stretnú pri stromčeku na pozícii 11.

3. Z

Iste ste si všimli, že svetu v poslednej dobe nehrozila žiadna zničujúca katastrofa ako napr. mimozemšťania, či neposlušné asteroidy. Toto teda výskumníci z NASA nemohli nechať len tak. Zvolali preto všetky kockaté mozgy, čo sa dali kúpiť a poďho na to.

Čo nikto načakal (ale my hej. . . :-), Sergejovi sa podarilo odhaliť separatistickú skupinu banditov, ktorej každý člen je označený štvorcom nejakého (vždy iného) prirodzeného čísla i (kde $i = 1, 2, \dots$). Keď sa títo banditi stretnú naraz na ulici, usporiadajú sa automaticky vzostupne podľa hodnoty svojich označení, takže tvoria nekonečný rad nejakým spôsobom sa striedajúcich čísel 0–9.

Keď sa už raz banditom podarí usporiadať sa, v priebehu niekoľkých okamihov sa dohodnú, na ktorom mieste v našom nekonečnom rade bude umiestnená zbraň Z , ktorá všetkých ľudí premení na kocky. Sergej je hlavička, a tak vám dokáže veľmi rýchlo zistiť a poskytnúť informáciu o mieste, na ktorom ukryli banditi zbraň (to vysleduje z ich správania – obyčajná psychológia). Ako však rýchlo a ľahko zistiť kód na odstavenie zbrane? Samozrejme tak, že zistíte, aká cifra sa na Sergejom určenom mieste nachádza. Tipovať nemôžete, máte len jeden pokus.

Uvedomte si, že oči celého sveta sa upierajú na vás!

Úloha: Ak napíšeme za seba štvorce prirodzených čísel $1 \dots \infty$, dostaneme rad zložený z čísel 0–9. Na vstupe je číslo k . Vašou úlohou je vytvoriť algoritmus, ktorý nájde cifru na k -tom mieste v tomto rade.

Príklad:

Vstup:

$k = 13$

Výstup:

4

(Pozn.: rad sa začína 149162536496481...)

¹ vtedy totiž (spolu) zabudnú najmenej – čím ďalej musí trpaslík zájsť, tým viac toho zabudne

4. Záhady Kiribatského rybolovu

Kiribatská vláda sa rozhodla zvýšiť príjmy štátu zavedením povolení na rybolov. Povolenia sa udeľujú nasledovne: Mapa Kiribati (ako isto viete) je rozdelená na $M \times N$ štvorcov. Pre potreby udeľovania povolení je každý z týchto štvorcov prehlásený buď za pevninu, alebo za more. Za 10 kiribatských bubákov môže ľubovoľný občan požiadať o povolenie na rybolov. Do žiadosti musí uviesť časť mapy, pre ktorú má toto povolenie platiť. Táto časť musí byť obdĺžnikového (resp. štvorcového) tvaru a má byť tvorená práve niekoľkými štvorcami mapy. Každá takáto oblasť je teda jednoznačne určená súradnicami štvorcov mapy v dvoch jej protiľahlých rohoch. Povolenie je občanovi udelené práve vtedy, keď oblasť, ktorú vyplnil do žiadosti, obsahuje iba more.

Väčšina Kiribatčanov rada loví ryby. Na druhej strane málokto z nich sa vyzná v kartografii. Takže väčšina občanov si vyzdvihne na úrade prázdnu žiadosť, vyplní do nej nejaké náhodné čísla a zájde s ňou späť na úrad. Pokiaľ dostane žiadosť, ide loviť ryby, pokiaľ nie, vyzdvihne si ďalšiu. A štát veselo zarába. Tak si to aspoň všetci vo vláde predstavujú.

Jediné, čo chýba, je program, ktorý by **veľmi rýchlo** vedel rozhodovať o žiadostiach, či ju treba prijať alebo zamietnuť. Na úradoch sa totiž čakajú také návaly, že nebude v silách úradníkov zvládnuť ich bez pomoci výpočtovej techniky.

Úloha: Priživte sa spolu s kiribatskou vládou na udeľovanie povolení na rybolov a napíšte program, ktorý bude prijímať a zamietajú žiadosti podľa vyššie uvedeného popisu. Program dostane vo vstupnom súbore mapu Kiribati v nasledovnom formáte: Prvý riadok obsahuje dve čísla M , N – počet riadkov a stĺpcov mapy. Ďalšie riadky obsahujú samotnú mapu, pričom j -ty znak v i -tom riadku je 1, ak je vo štvorci $[i, j]$ pevnina, resp. 0, ak je tam more.

Po spustení má váš program načítať mapu, prípadne si niečo dopredu pripraviť a potom čítať zo vstupu dôležité údaje žiadostí. Každý riadok bude obsahovať 4 čísla – súradnice dvoch protiľahlých rohov oblasti, o ktorú občan žiada. Program má vypísať **pridelit** alebo **zamietnut** podľa toho, či je v tejto oblasti len more alebo aj pevnina. Je dôležité, aby váš program vedel o ľubovoľnej žiadosti rozhodnúť čo najrýchlejšie.

Príklad:

Vstup:

Mapa:

8 10

0000000000

0000000010

0011000110

0111100000

0000110000

0000001000

0111000000

0000000000

Žiadosti:

1 1 2 3

3 3 3 3

8 10 1 1

1 7 4 6

Výstup:

pridelit

zamietnut

zamietnut

pridelit

5. Zablúdená správa

ERPAATNIC_PNOAJEEASJTSIOE_DVRENERM_EPDMR_TEK,TOSoy_ZME_AILUTZ_NVR_UZDIAAEAN
_ISTJIBITLY_NEDCIOVOHDRUU_HESMOA_PIR_NSK,TO_YCVHEKPOVYKSO_JRREAKCUNEE_DMIA
VJ_TTEO_CELIINYISNAT_AT_CSTKIOMVOSURBE_KSEYTCHHUZICJICOD_JIENC_OV_HTTTOSHDR

UU, _L_A_RUEF_AEKNTTAEKATUA_SA_LMCYELDCDEA_NITSAET_REE_A_USSP_MCH_OZALCAKHYKO
M_VEPKOD_OIEKN._

„*Táto správa musela určite zablúdiť,*“ povedalo si Ministerstvo Špionáže po tom, ako sa ich najlepším odborníkom na poli kryptografie túto správu nepodarilo rozlúštiť. A ešte k nej bol pribalený aj obrázok. . . „*No to si najskôr nejaké dieťa krátilo svoj voľný čas a nám vyrábalo starosti!*“

Úloha: Ale možno. . . čo ak predsa táto správa niečo hovorí? Keby sme aspoň vedeli, akým spôsobom bola táto šifra vyrobená. . . Znak ‘_’ označuje medzeru, nikde inde v texte medzery nie sú. Pokúste sa Ministerstvu pomôcť rozlúštením tejto správy a opísaním šifrovacej metódy.

