

Korespondenční Seminář z Programování

ZAČÁTEČNICKÁ KATEGORIE

32. ročník



KSP-Z

Listopad 2019

Dostává se vám do rukou zadání druhé série tohoto ročníku **KSP-Z**, neboli **Korespondenčního Semináře z Programování, Začátečnické kategorie**. Tuto sérii vydáváme k příležitosti Dne otevřených dveří na Matfyzu. **Úlohy ještě nejsou připravené k odevzdávání na webu, ale brzy budou.**

Zapojit se může každý středoškolák i základoškolák, řešit můžete začít i v případě, kdy jste se nezúčastnili první série. Ty nejúspěšnější z vás na jaře pozveme na týdenní soustředění, na kterém se toho spoustu dozvíte a zároveň si užijete hromadu neopakovatelné zábavy.

Termín série: 13. ledna v 8:00, praktické úlohy za třetinu bodů až do 20. ledna

Obsah série: 4 praktické úlohy (značené ) – K těmto úlohám je nutné napsat program (v libovolném vhodném jazyce), stáhnout si z našeho webu vstupní data a odevzdat odpovídající výstup. 2 teoretické úlohy (značené ) – U těchto úloh nás zajímá hlavně slovní popis řešení, ve kterém byste měli zdůvodnit jeho funkčnost a ideálně nás i přesvědčit o jeho efektivitě.

Odevzdávání: Přes web na adrese <https://ksp.mff.cuni.cz/z/odevzdej.cgi>



Zadání druhé série začátečnické kategorie 32. ročníku KSP

32-Z2-1 Prší 8 bodů

Už je tu zase podzim! Kevin seděl v pauze mezi vyučováním ve škole a koukal z okna na déšť bubnující do okrasné skalky před jejich školou. Skalka byla postavená v moderním stylu z různě vysokých bloků a brzy se v ní začaly tvořit různé velké louže a tůňky. Protože do odpoledního vyučování zbýval ještě dostatek času, tak se Kevin rozhodl spočítat si, kolik vody se celkově ve skalce zadrží.

Skalku si pro jednoduchost představme jako jednu linii bloků, kde každý blok je široký jeden decimetr a výšky bloků jsou také v celých decimetrech – skalku si tak můžeme představit jako na schématu níže. Na skalku prší a voda se zadržuje v prohlubních. Voda, která přeteče přes okraj, se vsákne a zmizí. Nás by zajímalo, kolik může skalka maximálně zadržet vody (v našem schématu jsme místa zadržení vody znázornili tečkami – skalka zvládne zadržet vodu v 5 čtverečcích o hraně jednoho decimetru, což je 5 litrů).

```
#
# . # . . #
##### . . ##
#####
```

Tvar vstupu: Skalku dostaneme zadanou jako posloupnost výšek jednotlivých bloků tak, jak stojí vedle sebe. Nejprve na prvním řádku vstupu bude číslo N udávající počet bloků a na druhém řádku se bude nacházet N celých nezáporných čísel oddělených mezerou udávající výšky bloků.

Tvar výstupu: Odpovězte jedním celým číslem, kolik litrů vody se ve skalce může maximálně zadržet.

Ukázkový vstup: *Ukázkový výstup:*

```
8
3 2 3 4 1 1 3 2
```

5

32-Z2-2 Turnaj hada 10 bodů

V pauze do odpolední výuky ještě Kevin se Zuzkou naplánovali turnaj ve hře had. Hráli klasického hada ve čtvercové síti – had má nějak dlouhé tělo, v každém kroku se hlavou posune o jedno políčko dál a současně se mu ocas také o jedno políčko zkrátí (tak, aby byl pořád stejně dlouhý).

Největší zábava je, když hada hraje více lidí najednou a navzájem si zahrazují cestu. Pokud totiž had po konci svého

kroku skončí hlavou v nějakém jiném hadovi (nebo i sám ve svém vlastním těle), tak umírá a do dalšího kroku zmizí. Speciálně se dva hadi mohou zabít i navzájem (třeba tím, že do sebe narazí hlavami).

Dostanete záznam stisků kláves jedné takové partie hada a vaším úkolem bude určit, po kolika krocích který z hadů zemřel (případně že i po dohrání všech kroků zůstal naživu).

Tvar vstupu: Na prvním řádku dostanete číslo H udávající počet hadů a číslo K udávající počet kroků. Poté bude následovat popis H hadů – každý popis začíná řádkem s jedním číslem D udávajícím délku hada, pak následuje řádek s D dvojicemi čísel popisujícími souřadnice políček hada od ocasu k hlavě – všechna čísla jsou oddělená mezerami, osy x a y rostou směrem doprava nahoru a hrací plocha není nijak omezená (had může zajet libovolně daleko).

Poté následuje řádek se řetězcem z K znaků l , p , n nebo d udávající pohyby hlavy hada doleva, doprava, nahoru a dolů (vzhledem k souřadnicové síti). Slibujeme vám, že vstup bude korektní (kostičky hada na sebe budou navazovat a kroky nebudou obsahovat otočku „čelem vzad“).

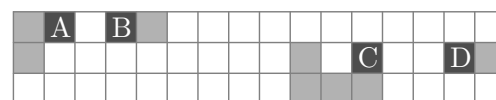
Tvar výstupu: Na H řádků vypište pro každého hada (ve stejném pořadí, jako na vstupu) po kolikátém kroku zemřel, případně řetězec NAZIVU, pokud přežil.

Ukázkový vstup:

```
4 5
3
0 1 0 2 1 2
ppppp
2
4 2 3 2
dlnll
5
9 1 9 0 10 0 11 0 11 1
ldlll
2
15 1 14 1
lllll
```

Ukázkový výstup:

```
NAZIVU
3
2
NAZIVU
```



Situace z příkladu vypadá jako na obrázku, tmavě jsou vyznačené hlavy. Had A jde sice doprava, ale těsně mu uhne ocas hada B a přežije, naopak had B nabourá do těla hada A a zemře. Had C nabourá do svého vlastního těla a had D projede v pohodě, jelikož před ním had C po umření zmizí.

32-Z2-3 Panika v chodbě 10 bodů

Najednou zazvonilo. „Jak to?!?“ podíval se Kevin vyděšeně na hodiny visící v chodbě a pak mu došlo, že se asi zastavily. A nebyl sám, koho nenadále zvonění zmátlo – na chodbě mezi dvěma třídami vznikl hotový zmatek, jak se každý snažil co nejrychleji dostat do své třídy.

Chodbu si můžeme představit jako úsečku se třídami v koncových bodech. Pro každého studenta máme zadanou počáteční pozici a to, kam a jak rychle utíká. Ale protože je to pořádný zmatek, tak do sebe studenti různě strkají a kdo spadne na zem, ten zaručeně přijde na hodinu pozdě.

Když rychlejší student předběhne pomalejšího, tak je pomalejší student odstrčen, spadne na zem a „vypadává“. Když se dva studenti srazí čelně, tak také „vyhrává“ ten rychlejší a pomalejší je vyřazen (slibujeme, že v našich vstupech nikdy nepoběží proti sobě dva studenti se stejnou rychlostí).

Vášim úkolem je zjistit, kolik studentů se dostane do svých tříd bez toho, aniž by je někdo vyřadil shozením na zem.

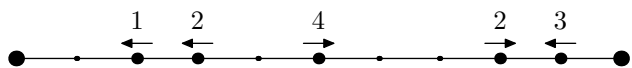
Tvar vstupu: Na prvním řádku dostanete dvě celá čísla D a S udávající délku chodby a počet studentů. Na dalších S řádcích pak vždy dostanete dvě celá čísla p a r udávající počáteční pozici studenta p (bude platit $0 < p < D$) a rychlost studenta r (pro kladné číslo běží student doprava, pro záporné číslo běží student doleva, nula se nevyskytne).

Tvar výstupu: Odpovězte jedním celým číslem udávajícím, kolik studentů se dostane na nějaký z konců chodby, aniž by byli někým vyřazeni.

Ukázkový vstup: *Ukázkový výstup:*

10 5	2
2 -1	
3 -2	
5 4	
8 2	
9 -3	

Příklad výše si můžeme představit jako následující obrázek:



Student nejvíce vlevo bude předběhnout studentem za ním. Student nejvíce vpravo nejdříve srazí studenta s rychlostí 2, než je sražen studentem s rychlostí 4. Zbylí dva studenti již do svých tříd doběhnou.

32-Z2-4 Opisoování v testu 12 bodů

Kevin se Sárou vběhli do třídy mezi posledními a hned za nimi vešla učitelka. Nesla testy z minulého týdne a tvářila se nezvykle přísně: „Myslím si, že více než půlka z vás navzájem opisovala,“ zahájila hodinu a pak jim ukázala, jak podobné si některé testy byly. Kevin neopisoval, ale rád by teď zjistil, kterou část testů měli všichni nejpodobnější.

Testy měly mnoho otázek s odpověďmi A až Z a tak si jedno vyplnění testu můžeme představit jako dlouhou posloupnost velkých písmen anglické abecedy. Kevin by chtěl

najít nejdelší úsek, ve kterém existuje více než $N/2$ testů, které jsou všechny vyplněné stejně (na tomto úseku).

Tvar vstupu: Na prvním řádku dostanete dvě čísla T a N oddělená mezerou udávající počet testů a počet otázek (délku testu). Na dalších T řádcích se pak budou nacházet řetězce N velkých písmen anglické abecedy.

Tvar výstupu: Na výstup vypište tři čísla oddělená mezerou určující nejdelší úsek, kde je více než polovina testů stejná. První číslo určuje pořadí první otázky v tomto úseku (číslováme otázky od nuly), druhé číslo určuje pořadí poslední otázky v tomto úseku a třetí číslo udává počet testů shodných na tomto úseku.

Ukázkový vstup: *Ukázkový výstup:*

5 8	1 7 3
XBAKEFGC	
ABAKEFGC	
ABAKEFGC	
ABAKEXXX	
ABAKEXXX	

Nejdelší společný úsek je BAKEFGC prvních tří studentů.

32-Z2-5 Asfaltéřský problém 12 bodů

Cestou domů šel Kevin přes náměstí, kde právě asfaltovali spoustu děr. Kevin se zamyslel, kolik děr by zvládl vyasfaltovat na jedno přejetí asfaltovacím vozem přes náměstí.

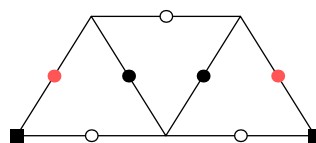
Náměstí si můžeme představit jako obdélník a díry v něm jako body se zadanými souřadnicemi (které nemusí být celočíselné). Asfaltovací vůz může jet přes náměstí jenom rovně z východu na západ (nemůže jet našikmo ani zatáčet) a vyasfaltuje přitom pruh široký D metrů.

Vymyslete algoritmus, který pro zadané náměstí s dírami a pro zadané D najde místo, ze kterého má asfaltovací vůz vyjet, aby jedním přejezdem vyasfaltoval co nejvíce děr.

32-Z2-6 Černobíločervená hra 14 bodů

Večer se Kevin sešel se svými kamarády a Petr přitáhl logickou hru, která mu právě přišla z jednoho zahraničního eshopu jako dárek pro malou sestru. Byla to trojúhelníková síť s barevnými korálky na hranách a dlouhým černobílým provázkem, který šel napnout přes hrany trojúhelníků.

Hra spočívá v tom, že jeden z hráčů rozestaví bílé, černé a červené korálky tak, že na každé hraně leží jeden, a pak určí startovní a cílový vrchol. Druhý hráč pak musí napnout provázek přes hrany trojúhelníků tak, aby se střídaly bílé a černé hrany (počáteční barvu si může vybrat libovolně), červené hrany je zakázané používat úplně. Cílem je samozřejmě najít nejkratší cestu ze startu do cíle. Vymyslete algoritmus, který takovou nejkratší cestu vzládne najít.



Například v této trojúhelníkové síti musíme na cestě mezi hranatými vrcholy objet jeden trojúhelník dokola, aby nám vyšlo střídání barev. Nejkratší cesta má tak délku 5.



KSP pro vás připravují studenti Matematicko-fyzikální fakulty Univerzity Karlovy.

Webové stránky:
<https://ksp.mff.cuni.cz/>

E-mail:
ksp@mff.cuni.cz

Diskusní fórum:
<https://ksp.mff.cuni.cz/forum/>

Chcete-li s námi komunikovat bezpečně, můžete si ověřit náš HTTPS certifikát – jeho SHA1 fingerprint je: E9:DB:EE:C6:62:BC:14:DE:09:E4:E8:97:DC:36:0E:87:B3:50:B0:01.