

Korespondenční Seminář z Programování

ZAČÁTEČNICKÁ KATEGORIE

35. ročník


KSP-Z


Prosinec 2022

Právě se díváte na leták třetí série 35. ročníku **KSP-Z**, neboli **Korespondenčního Semináře z Programování, Začátečnické** kategorie. Zapojit se může **každý středoškolák i základoškolák**, řešit můžete začít i v případě, kdy jste se nezúčastnili prvních sérií. Ty neúspěšnější z vás na jaře pozveme na **týdenní soustředění** (pokud to aktuální situace dovolí), na kterém se toho spoustu dozvíte a zároveň si užijete hromadu neopakovatelné zábavy.

Letos bude v KSP-Z **pět sérií po čtyřech úlohách** za celkem **220 bodů**. Pokud budete mít jakoukoliv otázku, neváhejte se zeptat. Kontaktní adresy najdete v patičce na konci letáku. Přejeme hodně štěstí!

Termín série: neděle 12. února ve 32:00 (tedy další ráno v 8:00), praktické úlohy za třetinu bodů až do 19. února

Obsah série: 3 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.

1 teoretická úloha (značená ) – U této úlohy 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/odevzdavatko/>



Zadání třetí série začátečnické kategorie 35. ročníku KSP

35-Z3-1 Vánoční ozdoby 8 bodů

Kevin se rozhodl, že si doma udělá vánoční výzdobu, a proto si objednal N svítících ozdob. Když konečně dorazily, rozbil je, celý natěšený, že si je pověsí na svůj vánoční stromek.

Jenže ouha. Ozdoby sice dorazily, ale všechny bez jakýchkoliv zdrojů, ze kterých by je mohl napájet. Co teď? Kdyby ozdoby reklamoval, do Štědrého dne mu nedorazí. A za stromeček, který ani nesvítí, se mu všichni kamarádi budou smát.

Vzpomněl si však, že v jeho ulici je malý obchůdek s elektronikou. Třeba by tam mohli mít něco, čím půjde světélka napájet. Když přišel do obchodu, zajásal – prodávají zde tři různé typy zdrojů, kterými půjde ozdoby napájet. Když ale uviděl, kolik zdroje stojí, jásoť jej přešel. Rozhodl se proto, že je nakoupí co nejlevněji to půjde.

Jenže to není tak jednoduché. Každá z jeho N ozdob má totiž nějaké minimální a maximální tolerované napětí. Pokud by k ní připojil zdroj o napětí nižším než je to minimální, ani by se nerozsvítila. Pokud by připojil vyšší napětí, světélko by shořelo.

Kevin ke každé ozdobě tedy potřebuje koupit zdroj tolerovaného napětí. Jak to má udělat, aby zaplatil za zdroje co nejméně?

Toto je praktická open-data úloha. V odevzdávacím systému si necháte vygenerovat vstupy a odevzdáte příslušné výstupy. Záleží jen na vás, jak výstupy vyrobíte.

Formát vstupu: Na první řádce dostanete číslo N – počet ozdob, které si Kevin objednal. Následují tři řádky. Na i -té z nich se nachází mezerou oddělená dvě čísla U_i a C_i – napětí a cena i -tého typu zdroje. Nakonec následuje N řádek. Na j -té z nich se nachází dvě čísla U_j^{min} a U_j^{max} – minimální a maximální napětí j -té ozdoby.

Formát výstupu: Na výstup napište mezerou oddělená čtyři čísla – M_1 , M_2 , M_3 a C , kde M_i je počet zdrojů i -tého typu, které má Kevin nakoupit, a C je celková cena, kterou Kevin za zdroje zaplatí. Slibujeme, že nějaké řešení vždy existuje.

Ukázkový vstup:

5
5 100
10 200
20 300
4 20
5 5
11 20
20 30
1 25

Ukázkový výstup:

3 0 2 900

35-Z3-2 Padající piškvorky 11 bodů

Keď mal Kevin nakúpené zdroje a rozvešané všetky nové ozdoby, spomenul si, že vlastní aj netriviálne množstvo ozdôb z rokov minulých. Boli odložené na povale v krabiciach, a z plochy, ktorú zaberali, bolo jasné, že sa mu všetky rozvešať nepodarí. Zavolať si teda Sáru, aby mu pomohla vybrať, ktoré sa najviac hodia k tým novým.

Popri vyberaní ozdôb z krabíc však narazili na starú hru Piškvorky. Boli to také tie vertikálne piškvorky, pri ktorých sa hráči striedajú vo vhadzovaní svojich žetónov do stĺpcov, a každý žetón spadne vplyvom gravitácie až na najnižšie neobsadené pole svojho stĺpca. Okamžite zabudli na pôvodný problém a začali hrať. Odohrali toľko partii, že si už nepamätajú, koľko to bolo.

Hracia plocha mala S stĺpcov, a bola naozaj vysoká – dokonca tak, že sa im nikdy nepodarilo ju zaplniť až po vrch predtým, než jeden z nich vyhral, alebo ich to prestalo baviť. Vždy, keď niekto vyhral, celú plochu vyprázdnil, a pokračovali v novej hre, pričom začínal hráč, ktorý predtým prehral. Aby boli hry trochu netriviálne, Kevin navrhol, nech vyhráva hráč, ktorý ako prvý nazbiera zo svojich žetónov tvar hrocha - štvorec veľký aspoň 3×3 (namiesto štandardného pravidla o štyroch žetónoch v rade).

Keď ich hra prestala baviť, rozhodli sa spočítať, kto z nich je celkový víťaz. Zistili ale, že si presne nepamätajú, koľkokrát vyhral. Pamätajú si jedine, že poslednú hru vyhral Kevin. Našťastie si Sára počas celej hry značila, do ktorého stĺpca niekto vhodil žetón (ale nie, kto ho tam vhodil). Dokopy sa jej v poznámkach nazbieralo až N zoradených

záznamov, ktoré ale nie sú oddelené po jednotlivých hrách. Teraz by ich zaujímalo, kto kolkokrát vyhral, a v ktorom ťahu každé víťazstvo nastalo.

Toto je praktická open-data úloha. V odevzdávacím systéme si necháte vygenerovať vstupy a odevzdáte príslušné výstupy. Záleží len na vás, jak výstupy vyrobíte.

Formát vstupu: Na vstupe dostanete najskôr jeden riadok s dvomi číslami oddelenými medzerou: N , počet záznamov v zošite (teda celkový počet odohraných ťahov), a S , počet stĺpcov hracej plochy. Nasleduje N čísel v rozsahu od 1 do S , každé na vlastnom riadku, sú to čísla stĺpcov, do ktorých boli postupne vhadzované žetóny.

Formát výstupu: Na výstup vypíšete najskôr všetky víťazstvá zoradené podľa ťahu, v ktorom nastali: každé víťazstvo nech je na jednom riadku, a tvorí ho číslo ťahu (číslujeme od 1) a meno víťaza (Kevin alebo Sara) oddelené medzerou. Napríklad ak by nejaká hra skončila víťazstvom Sáry v 136. zápise, tak vypíšete 136 Sara Na posledný riadok vypíšete meno celkového víťaza, alebo remiza, ak mali obaja dokopy rovnako veľa výhier.

Ukážkový vstup:

```
17 10
1
9
2
8
3
7
1
8
2
7
3
9
2
8
3
9
1
```

Ukážkový výstup:

```
17 Kevin
Kevin
```

35-Z3-3 Substituční šifra 12 bodů

Mezi hrochy obojživelnými a hrošíky liberijskými panuje, jak už to tak u příbuzných bývá, odjakživa velká rivalita. Dříve hrochům stačila jejich velikostní a silová převaha, aby si vydobyli to nejlepší místo na rochnění v bahně, ale hrošiči jsou vychytralí a začínají využívat sofistikovaných válečných taktik. Jednou z nich je šifrování – své bojové plány si posílají v tajných zprávách a hroši pak neví, z jaké strany zaútočí.

Proto si hroši přivolali kryptografa Kevina, aby zachytené zprávy rozluštil. Není v tom ale sám, špiónka Sára se úspěšně infiltrovala mezi hrošíky, ale samozřejmě nemůže Kevinovi původní text zpráv nebo šifrovací klíč poslat bez toho, aby se prozradila. Jediné, co se jí podařilo vyslat ven, jsou počty jednotlivých písmen abecedy v původním znění

každé ze zpráv a také informaci, že hrošiči šifrují tak, že prostě nahradí každé písmeno abecedy jiným (substituční šifrou).

Toto je praktická open-data úloha. V odevzdávacím systéme si necháte vygenerovat vstupy a odevzdáte příslušné výstupy. Záleží jen na vás, jak výstupy vyrobíte.

Formát vstupu: Na prvním řádku dostanete 26 čísel, každé reprezentující počet výskytů jednoho písmene abecedy v původním textu. (První číslo je pro A, druhé pro B, atd.) Můžete počítat s tím, že počty výskytů písmen, které se v textu vyskytují, jsou unikátní. Na dalším řádku dostanete číslo N a poté N řádků zašifrované zprávy. Ta obsahuje pouze malá písmena anglické abecedy a bílé znaky (white-space – mezery, tabulátory a odřádkování).

Formát výstupu: Celý text rozšifrované zprávy. Bílé znaky šifra nemění, proto je také nechte nezměněné. Stejně jako u vstupu i výstup by měl obsahovat pouze malá písmena anglické abecedy a bílé znaky.

Pozn.: Autor úlohy neručí za biologickou přesnost této úlohy. Jestli umí hroši a hrošiči šifrovat, se ptejte zoologů, ne informatiků.

Ukážkový vstup:

```
5 4 3 2 1 [21krát 0]
1
ccacccl1bbbbbmm
```

Ukážkový výstup:

```
aaaaaaddbbbccc
```

35-Z3-4 Vánoční stromek 13 bodů

Když už měl Kevin nakoupené zdroje ke svým ozdobám, rozvěsil je všechny po stromě. S velkolepým rozsvícením stromečku by ale rád počkal na své kamarády a Štědrý den. Proto světélka zatím nezapojoval do zásuvky, už si však natahal prodlužovačky. Světélka a prodlužky pozapojoval do sebe, až mu zbyla jen jedna prodlužovačka, kterou pak stačí zapojit do zásuvky a celý stromeček se rozzáří. Formálně bychom řekli, že prodlužovačky tvoří zakořeněný strom (nikoliv vánoční), kde listy jsou ozdoby a vnitřní vrcholy jsou jednotlivé prodlužovačky.

Když však na Štědrý den stromeček zapojil, začaly svítit jen některé žárovky. Chvilí přemýšlel, čím to může být – vždyť ozdoby jsou úplně nové. Náhle mu to došlo – každá prodlužovačka má totiž maximální proud, který jí může procházet. Pokud ozdoby zapojené v této prodlužovačce (klidně i přes jiné prodlužovačky) odebírají větší proud, než je toto maximum, pak se pořádně nerozsvítí.

Kevin chce nyní co nejrychleji popřepojovat zařízení do zásuvek, aby stromeček svítil celý. V každém kroku si vybere nějakou prodlužovačku a všechno, co je do ní zapojeno, popřepojuje do zásuvek (těch má neomezeně mnoho). Kamarádi a dárky už ale čekají, takže by to chtěl udělat na co nejméně kroků. Jak na to?

Toto je teoretická úloha. Není nutné ji programovat, odevzdává se pouze slovní popis algoritmu. Více informací zde: <http://ksp.mff.cuni.cz/viz/tinfo>