

Korespondenční Seminář z Programování

ZAČÁTEČNICKÁ KATEGORIE

35. ročník


KSP-Z


Červenec 2022

Právě se díváte na leták první série 35. ročníku **KSP-Z**, neboli Korespondenčního Semináře z Programování, Začátečnické kategorie. Jedná se o korespondenční soutěž spočívající v řešení jednodušších programátorských úloh, které vychází během školního roku. Zapojit se může každý středoškolák i základoškolák, ty úspěšnější z vás pak na jaře pozveme na týdenní soustředění, na kterém se toho spoustu dozvíte a zároveň si užijete hromadu neopakovatelné zábavy. Pokud budete mít jakoukoliv otázku, neváhejte se zeptat. Kontaktní adresy najdete v patičce na konci letáku. Přejeme hodně štěstí!

Letos bude v KSP-Z **pět sérií** po **čtyřech úlohách** za celkem **220 bodů**.

Termín série: 23. října 32:00 (tedy další ráno v 8:00), praktické úlohy za třetinu bodů až do 30. října

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 první série začátečnické kategorie 35. ročníku KSP

35-Z1-1 Slovíčka 8 bodů

V Kevinově třídě píší na závěr roku test z angličtiny na překlad slovíček. Kevin ví, že není moc dobrý na jazyky, proto se je učil nazpaměť dlouho po nocích. Teď má pocit, že by dokázal vyjmenovat všechna anglická slovíčka, která byla na seznamu na naučení. Není si už ale vůbec jistý, jestli něco dovede přiřadit k českému překladu.



Hned první otázka ho dostala. Jak se řekne anglicky *dikobraz*? Kevin netušil. Vzpomněl si ale, že *dikobraz* byl na páté stránce slovíček nahoře, kousek nad ním bylo *prase*, což je anglicky *pig*, a kousek pod ním *nosorožec*, což je *rhinoceros*. Jelikož byl slovníček seřazen podle abecedy, došlo mu, že *dikobraz* se musí přeložit na něco, co je podle abecedy mezi *pig* a *rhinoceros*. Kolik takových slovíček vůbec je?

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 prvním řádku jsou dvě čísla: počet slovíček ve slovníku N a počet otázek v testu Q . Na následujících N řádcích je jedno slovo, které sestává z nejvýše 20 znaků malé anglické abecedy. Tato slova nemusí být nutně abecedně seřazena. Na dalších Q řádkách jsou vždy dvě různá slova ze slovníku oddělena mezerou.

Formát výstupu: Pro každý z Q dotazů vypíšete na samostatný řádek jediné číslo – počet slovíček, která jsou v abecedním uspořádání mezi dvojicí slovíček na vstupu.

Ukázkový vstup:

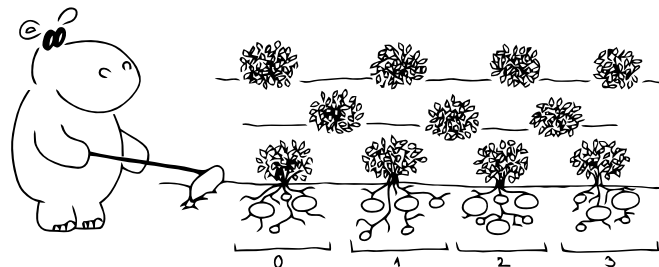
7 2
platypus
pig
porcupine
rhinoceros
rabbit
possum
panda
pig rhinoceros
rabbit porcupine

Ukázkový výstup:

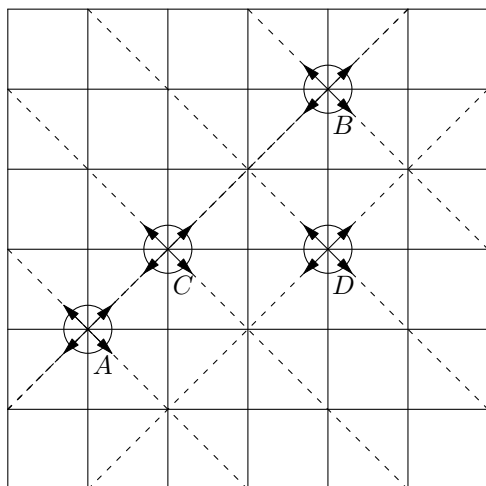
4
1

35-Z1-2 Kanón na vrabce 10 bodů

„Už toho mám vážně dost! Proč jdou jen po mně?“ zalamentoval Kevin, když přišel na svou zahrádku. Letní počasí v něm se Sárrou probudilo zápal pro zahradničení a začali se společně předhánět v pěstování slunečnic. Jenže ke Kevinově nelibosti se tato soutěž zalíbila i vrabčům, kteří se teď na jeho zahradě napásají do sytosti. Co teď, aby nevypadal před kamarádkou trapně?



Naštěstí je tu ještě Zuzka, která přichází s poněkud ambiciózním plánem, takovým kanónem na vrabce. A to doslova. Ze skladu vyřazeného vojenského materiálu nosí jeden kanón za druhým a připevňuje je ke sloupům. „Teď už si na tebe nepřijdou“, povídá spokojeně. Kevin je nedůvěřivý. Výhra ho láká, ale bojí se, jestli v tomto rozestavení nemůže jeden kanón trefit druhý a tím způsobit explozi, která zničí nejen slunečnice, ale i celou zahradu. Pomůžete mu s rozhodováním?



Na obrázku jsou v bodech A až D připevněny kanóny. Každý kanón umí střílet do čtyř směrů diagonálně. Ohrožují se tedy kanóny A a C a také B a C . Kanóny A a B se neohrožují, protože je mezi nimi kanón C .

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 prvním řádku se nacházejí mezerou oddělená čísla W a H udávající po po řadě šířku a výšku Kevinovy zahrady a číslo N . Na následujících N řádcích jsou mezerou oddělené horizontální a vertikální souřadnice kanónů v mezích zahrady. Každý kanón míří do čtyř směrů po diagonálách a dostřelí na neomezenou vzdálenost.

Formát výstupu: Na jediný řádek vypište mezerou oddělená čísla libovolných dvou kanónů, které se ohrožují. Kanóny číslováme od nuly v pořadí, jak byly na vstupu. Slibujeme, že alespoň jedna taková dvojice existuje.

Ohrožují se dvojice kanónů ležící na stejné diagonále, mezi kterými neleží žádný další kanón.

⬆️ **Lehčí varianta (za 2 body):** V prvních dvou vstupech slibujeme, že se ohrožuje právě jedna dvojice.

Následující vstup odpovídá situaci na obrázku:

<i>Ukázkový vstup:</i>	<i>Ukázkový výstup:</i>
6 6 4	1 2
1 2	
4 5	
2 3	
4 3	

📅 35-Z1-3 Plánování schůzky 12 bodů

Organizátoři KSP by se rádi sešli a vymýšleli úlohy do dalších sérií. Jenže Kristýna může jen od tohoto pondělí do pátku, Vašek je na dovolené, a tedy může až od soboty, Jirka celé prázdniny cestuje, takže může pouze ve středu ...

Je obecně známo, že čím více hlav, tím kvalitnější úlohy. Organizátoři by se proto rádi sešli v co největším počtu. Jenže i po opakovaném ujasňování si, kdo kdy může, se stále dohadují, který den je nejvhodnější. Budou potřebovat vaši pomoc!

Každý z organizátorů může na schůzku dorazit pouze v některých po sobě jdoucích dnech určených prvním a posledním volným dnem. Vaším úkolem je najít den, kdy může co nejvíce organizátorů. V případě, že je takových dnů více, vypište první z nich, protože tam je největší šance, že nikomu nepřibude žádná další neplánovaná událost.



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 prvním řádku je jedno číslo: počet organizátorů N . Následuje N řádků popisujících jednotlivé organizátory. Na každém z nich je dvojice čísel – první a poslední den, kdy se daný organizátor může účastnit schůzky. Pro jednoduchost dny značíme čísly od jedné.

Formát výstupu: Vypište na jeden řádek dvě čísla oddělená mezerou – označení prvního dne, kdy se může sejít co nejvíce organizátorů, a počet, kolik se jich daný den může sejít.

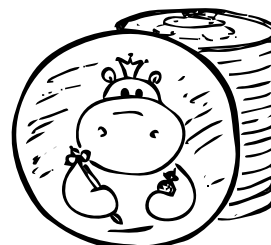
⬆️ **Lehčí varianta (za 6 bodů):** V prvních třech vstupech platí, že existuje den, kdy se mohou sejít všichni.

Pozor, některé vstupy obsahují až 10^9 dní.

<i>Ukázkový vstup:</i>	<i>Ukázkový výstup:</i>
4	4 3
1 5	
3 10	
4 9	
6 8	

💡 35-Z1-4 Šetření na hlídačích 14 bodů

Kevin si o prázdninách vydělal spoustu peněz, které teď potřebuje bezpečně uschovat. A protože není troškař, rozhodl se, že si po vzoru Strýčka Skrblíka postaví velkou kasu na peníze. Takovou kasu je však potřeba hlídat, a kdyby měl Kevin platit příliš mnoho hlídačů, z jeho bohatství by brzy nic nezbylo.



Rozhodl se tedy, že zjistí, kolik nejméně hlídačů si musí najmout. Už se mu povedlo napsat program, který dostane na vstupu počet hlídačů K , načtež spustí velmi složitou simulaci. Po hodině hučení počítač oznámí, zdali je zadaný počet hlídačů dostatečný.

Nyní se Kevin chystá program opakovaně pouštět s různými hodnotami K , dokud nenajde tu nejmenší možnou, kterou program označí jako dostatečnou. Jenže zkoušet každou


hodnotu by mohlo trvat celou věčnost. Naštěstí si Kevin vzpomněl na binární vyhledávání.¹ Kdyby věděl, že L hlídačů je ještě málo, ale H hlídačů už stačí, pustí svůj program s průměrem těchto hodnot $M = (L + H)/2$. Pokud je M málo, bude dále prohledávat interval od M do H . Pokud je to naopak dost, bude prohledávat interval od L do M . V obou případech se počet hodnot, které musí vyzkoušet, zmenší na polovinu, takže hledané optimum najde poměrně rychle.

Kevin si již myslel, že má vyhráno, když tu našel ve svém plánu háček. Neví, jak binární vyhledávání nastartovat. Počáteční volba pro L je jednoduchá, stačí použít $L = 0$, ale s jakou hodnotou pro H má začít?


Pomozte Kevinovi: Vymyslete, jak jeho algoritmus binárního vyhledávání upravit, aby se vypořádal s tím, že nemá zadaný horní limit H . Snažte se, aby program se simulací bylo potřeba spustit co nejméněkrát vzhledem k optimálnímu počtu hlídačů N .

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

Praktický kurz programování

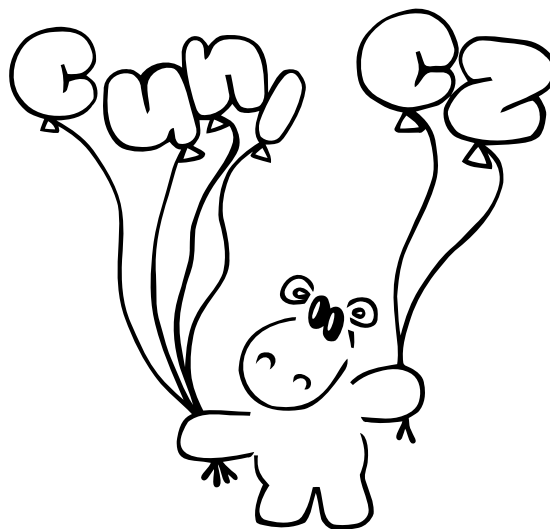
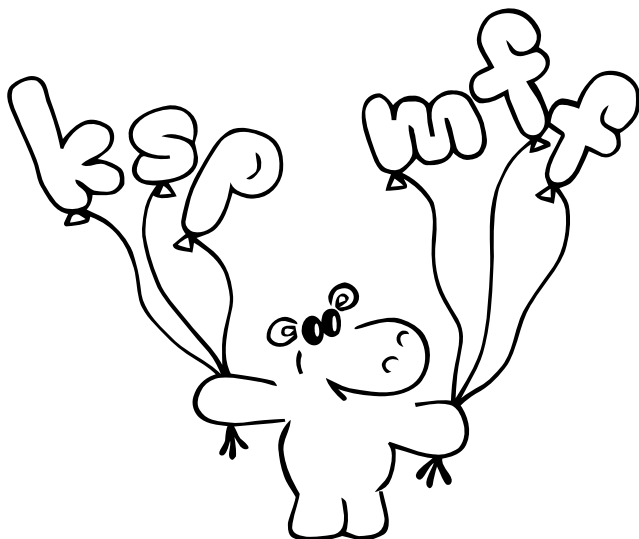
 Pokud Tě lákají praktické úlohy, ale ještě neumíš žádný programovací jazyk, můžeš se podívat na náš Základní kurz programování, kde se můžeš naučit základy Pythonu: <https://ksp.mff.cuni.cz/kurz/>.

Zdrojáky praktických úloh

 Řešení praktických úloh může být ze začátku složité. Velmi často i nějaká triviální technická chyba ve zdrojovém kódu programu může znamenat, že program vrací špatný výsledek – a některé chyby se ze začátku špatně hledají. Proto Ti nabízíme možnost poslat zdrojový kód programu nějaké úlohy na adresu zdrojaky@ksp.mff.cuni.cz, kde se Ti pokusíme poradit. Do emailu prosím připiš:

- Jakou úlohu by měl program řešit.
- Slovní popis, co by měl program podle Tebe dělat.

Před termínem série Ti nemůžeme radit s algoritmem, ale pomůžeme s odladěním zdrojáku. Po termínu série pak můžeme poradit i s návrhem algoritmu – získáš tak znalosti do dalších sérií.



¹ <http://ksp.mff.cuni.cz/viz/kucharky/binarni-vyhledavani>