

Korespondenční Seminář z Programování

ZAČÁTEČNICKÁ KATEGORIE

34. ročník


KSP-Z


Prosinec 2021

Právě se díváte na leták třetí série 34. 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 dvou sérií. Ty nejuspěšnější 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 13. února ve 32:00** (tedy další ráno v 8:00), praktické úlohy za třetinu bodů až do 20. 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 34. ročníku KSP

34-Z3-1 Otočená morseovka **8 bodů**

Kevin chce poslat Zuzce morseovkou zprávu do její rodné země. Zuzka je ale z Izraele, kde se mluví hebrejsky. Háček je v tom, že v hebrejštině se čte zprava doleva, nikoli zleva doprava jako v evropských jazycích. Kevin ale zapomněl, na kterém směru se dohodli, a teď mu je trapné se na to znovu ptát. Proto se rozhodl napsat program, který mu ověří, že se zpráva po přeložení do morseovky bude číst stejně z obou stran.

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

Vstup: Na prvním řádku vstupu dostanete číslo N určující počet zpráv. Pak následuje N řádků, každý obsahuje jednu zprávu. Zpráva je řetězec čísel a velkých písmen bez diakritiky. Zpráva neobsahuje žádné mezery ani tabulátory.

Výstup: Výstup bude N řádků, kde na každém řádku bude buď slovo ANO, pokud se zpráva přeložená do morseovky čte z obou stran stejně a jedná se tak o palindrom, a NE, pokud není.

Zde je tabulka s Morseovou abecedou:

"A": ".-.",	"B": "-...",	"C": "-.-.",
"D": "-..",	"E": ". .",	"F": ".-.-.",
"G": "--.",	"H": ". . . .",	"I": ". . .",
"J": ".-.-.-",	"K": "-.-.",	"L": ".-.-.",
"M": "--",	"N": "-.",	"O": "---",
"P": ".-.-.",	"Q": "--.-.",	"R": ".-.-.",
"S": ". . . .",	"T": "-.",	"U": ".-.-.",
"V": ". . . -",	"W": ".-.-",	"X": "-.-.-",
"Y": "-.-.-",	"Z": "--..",	"1": ".-.-.-.-",
"2": ".-.-.-.-",	"3": ".-.-.-.-",	"4": ".-.-.-.-",
"5": ".-.-.-.-",	"6": "-.-.-.-",	"7": "--.-.-",
"8": "-.-.-.-",	"9": "-.-.-.-",	"0": "-.-.-.-

Ukázkový vstup:

5
9K73K1
9FL1
Q05PP50Y
YVV4Q7TVR
TW1YQ9GT

Ukázkový výstup:

ANO
ANO
ANO
NE
ANO

34-Z3-2 Tečkový displej **10 bodů**

Kevin jede vlakem na návštěvu k babičce. Cesta je dlouhá, a tak se rozhodl, že se trochu prospí. Teď se probudil a došlo mu, že neví, kde je! Má spoustu věcí a potřeboval by se připravit před vystoupením, jenže věci jsou těžké, takže se nechce připravovat zbytečně. Proto by potřeboval rychle zjistit, jak se jmenuje příští zastávka.

Ve vlaku je tečkový displej, který by měl zobrazovat název příští zastávky. Bohužel je už hodně starý, a tak některé diody nesvítí, i když by měly. Kevin je ale opravdu zoufalý a rozhodl se, že se pokusí název rozluštit z toho, co vidí. Zná přece jména všech zastávek, třeba bude mít štěstí a jednoznačně pozná, která zastávka se na displeji zobrazuje.

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

Vstup: Na prvním řádku dostanete čtyři přirozená čísla: $R \leq 100$, $S \leq 100$, $N \leq 100$, $M \leq 20$. Číslo R udává počet řádků displeje, S udává počet sloupců. Displej je tedy reprezentovaný mřížkou $R \times S$ diod. Diody, které svítí, jsou označeny znakem x, nesvítící diody jsou označeny tečkou. Za prvním řádkem následuje N názvů zastávek, jak by byly zobrazeny na plně funkčním displeji. Potom následuje M porouchaných displejů, které se Kevin snaží rozluštit. Pro každý porouchaný displej určete, jestli je jednoznačné, jakou zastávku ukazuje. Vždy platí, že displej se pokouší zobrazit nějaký z N názvů zastávek, až na vadné nesvítící diody.

Výstup: Výstup bude mít M řádků. Pro každý porouchaný displej vypište jeden řádek ANO, pokud je zastávka na displeji jednoznačná. V opačném případě vypište NE.

Ukázkový vstup:

```
4 7 2 3
...x...
..x.x..
.xxxxx.
x.....x
xx...xx
x.x.x.x
x..x..x
x.....x
.....
..x.x..
...x...
x.....
.....
..x.x..
...xx..
.....
.x.....
..x.x..
...x...
.....x
```

Ukázkový výstup:

```
NE
ANO
ANO
```

Každá veličina je neprázdná posloupnost malých písmen anglické abecedy. Můžete se spolehnout, že všechny části vzorce jsou odděleny právě jednou mezerou (nezávisle na tom, jakou matematickou operací jsou provázané).

Výstup: Na výstup vypíšete posloupnost vzorců ve stejném formátu, jako na vstupu. Ta nám říká, v jakém pořadí používat vzorce, abychom ze zadání spočítali X . Vzorec můžeme použít, známe-li hodnoty obou veličin na pravé straně od rovná se. Tím se dozvíme hodnotu veličiny na levé straně. Jinak než použitím zadaných vzorců se nové hodnoty veličin zjistit nedají. Nejsou tedy povoleny substituce ani jiné úpravy (to víte, pravidla olympiády jsou striktní).

Posloupnost nemusí být nejkratší, ale každým vzorcem musíme spočítat něco nového, tedy jeho levou stranu ještě neznáme.

Ukázkový vstup:

```
3 6 z
a b c
d = e + fn
fn = a - d
g = z / a
z = fn + d
d = e - b
e = a * c
```

Ukázkový výstup:

```
e = a * c
d = e - b
fn = a - d
z = fn + d
```

34-Z3-3 Tabulky a olympiáda 12 bodů

Další ročník fyzikální olympiády je tu. Normálně by se Kevin se Sárou nemohli dočkat, ale letos to je jiné. Kvůli věku se již nemohou účastnit, a tak alespoň pomáhají s organizací. „Opravovatelé kontrolují řešení účastníků, ale kdo kontroluje opravovatele?“ rozčiluje se Kevin. „To je pravda, co když někoho neprávem připravíme o body?“ přitakává Sára.

Pomozte naší dvojici a napište program generující správná řešení fyzikálních úloh, máte-li k dispozici matematicko-fyzikální tabulky.

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

Vstup: Na prvním řádku dostanete mezerou oddělená čísla N a M a veličinu X . Na druhém řádku je mezerami oddělených N veličin, které se vyskytují v zadání úlohy. Předpokládáme tedy, že hodnoty těchto veličin jsou zadane a naše výpočty z nich mohou vycházet. Na následujících M řádcích jsou vzorce ve tvaru $a = b \circ c$, kde a , b i c jsou veličiny a \circ je $+$, $-$, $*$ nebo $/$.

34-Z3-4 Velikost plachet 14 bodů

Kevin se nachomýtl k práci v lodní dílně. Jeho úkolem je pomoci s přípravou stěžňů na jednotlivé lodě. U každého stěžně známe jeho výšku S . Dále jsou k dispozici ráhna s délkami R .

Na stěžň S s ráhnem R se dá pověsit plachta velikosti na nejvýš $S \times R$. Čím větší je plachta, tím více pobere větru a loď bude rychlejší, navíc z nějakého důvodu dílna nejvíce vydělává na plachtovině. Proto musí Kevin vymyslet algoritmus přiřazující ráhna ke stěžňům z jedné dodávky tak, aby součet velikostí plachet byl co největší. Protože nechce být vyhozen, musí vedoucímu dokázat, že jeho algoritmus vždy najde nejlepší řešení. Pomozte Kevinovi vyřešit tento problém.

Navrhněte algoritmus, který dostane délky stěžňů a ráhen a přiřadí je k sobě tak, aby celková plocha plachet, které se na ně natáhnou, byla co největší. Součástí řešení musí být i důkaz, že lépe už to nejde.