

Abecední pořadí



Na vstupu programu bude (na samostatných řádcích)

- celé číslo `n` (od 1 do 1 000),
- `n` řádků, na každém jedno slovo (dlouhé od 1 do 100 znaků).

Slova budou obsahovat jen písmena anglické abecedy. Stejné slovo se může objevit vícekrát.

Úkolem je vypsát, zda jsou (`ordered`) nebo nejsou (`unordered`) slova seřazena abecedně. V pořadí slov musí platit, že

1. ze slov, která začínají stejně, ale jedno je delší (obsahuje uvnitř sebe to druhé), **má přednost to kratší** a
2. budou-li se dvě slova lišit jen velikostí písmene, **mají přednost malá písmena** před velkými.

První pravidlo je přednější – vizte vysvětlení příkladu #1.

Platné řazení může být vzestupné (od A do Z) i sestupné (od Z do A). Pozor: dvě uvedená pravidla ale mají vždy stejnou přednost – vizte vysvětlení příkladu #2.

Příklad #1

Vstup

```
6
Mark
marksman
moon
Moon
moonwalk
moonwalk
```

Výstup

```
ordered
```

Vysvětlení

`marksman` sice začíná malým písmenem, ale je delší než `Mark`. Pravidlo č. 1 je přednější a proto je `Mark` první. Totéž platí u `Moon` a `moonwalk`. Mezi `moon` a `Moon` rozhoduje pravidlo č. 2.

Příklad #2

Vstup

```
7
zero
Isaac
IP
iphoneSE
iPhoneSe
internet
beta
```

Výstup

```
ordered
```

Vysvětlení

Slova jsou seřazena abecedně sestupně. Na přednosti pravidel to však nic nemění: `IP` je před `iphone`, protože je kratší – to odpovídá pravidlu č. 1. `iphoneSE` je před `iPhoneSe`, protože má malé písmeno (`p`) na dřívější (více vlevo) pozici – podle pravidla č. 2. Kdyby byla slova uvedena v opačném pořadí, výstupem by bylo `unordered`.

Veškeré výukové materiály se nachází na <https://github.com/delta-cs/seminar>.

V případě, že vám řešení nebude uznáno a nebudete vědět proč, navštivte nejdříve [stránku s technickými pravidly](#) a ujistěte se, že program všechna splňuje.