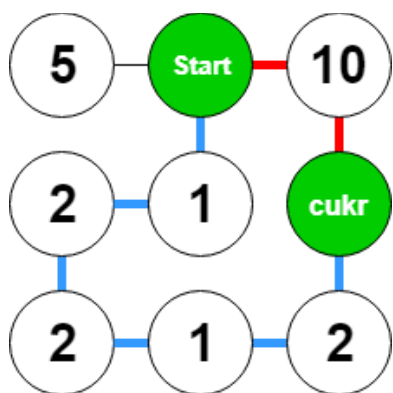


Kdysi dávno, v jedné předaleké galaxii...

Na planetě Quadient přistála kolonie mravenců (tzv. astroantů), kteří mají za úkol vybudovat základnu pro další průzkumné mise. Spolu s astroanty byla na Quadient vyslána i zásobovací loď plná kostek cukru, ale ta kvůli vlivu erupcí musela přistát na jiném místě. Jedinou šancí na přežití je pro astroanty najít cestu k zásobám dostatečně rychle, protože bez cukru jsou schopni přežít pouze 1 minutu. Naštěstí mají astroanti při ruce satelitní snímky celé planety a vědí, kde se přesně zásobovací loď nachází. Místní krajina je ovšem velmi rozmanitá, v některých oblastech je lehké prostupná a v jiných naopak skoro vůbec. Chtějí-li astroanti přežít, musejí se rychle rozhodnout, kudy se k cukru vydat, aby dokázali přežít.

Planetu si lze představit jako ortogonální a vždy čtvercové bludiště rozdělené do několika oblastí. Každá oblast má 1-4 přechody do jiné oblasti a časovou náročnost, jak dlouho trvá danou oblast projít (v milisekundách). Vaším úkolem je naimplementovat algoritmus, který **ze zadané mapy oblastí dokáže najít cestu astroantů k cukru v rozumném čase**. Rozumným časem je myšleno to, že součet času běhu algoritmu a času potřebnému na průchod přes danou cestu nepřekročí 1 minutu (než mravenci umřou na nedostatek sacharidů). Čas potřebný na průchod je suma náročností všech oblastí přes které se prochází. Mapy, které je potřeba vyřešit budou mít velikost 300x300 oblastí, nicméně bude jen dobře, pokud Váš algoritmus bude schopen zpracovat v dostatečně rychlém čase i rozlehlejší mapy.

Uvedeme příklad, pro zjednodušení s mapou 3x3 (na mapě 300x300 by se to hůře demostrovalo):



Pro tuto mapu existují 2 možné cesty:

- červená cesta trvající 10ms
- modrá cesta trvající 8ms

Váš algoritmus by měl být schopen najít alespoň jednu z těchto cest - ideálně tu nejlepší možnou. V tomto případě je ideální nalézt modrou cestu a to do 59992ms (1min-8ms), případně lze nalézt i červenou (na tu je ovšem vzhledem k náročnosti trasy méně času - konkrétně 59990ms (1min-10ms)).

Pro testování vašeho algoritmu můžete využít jednoduché REST API se 2 endpointy. **Použití API není nutnou podmínkou pro účast v soutěži**, pouze Vám může usnadnit testování vašeho algoritmu. Úlohu lze tedy bez problémů řešit i bez použití API. Quadient negarantuje, že endpointy poběží 24/7, nicméně pokud byste zaznamenali výpadky, kontaktujte nás prosím na e-mail go@quadiantevents.cz a my se je pokusíme co nejdříve vyřešit. Oba endpointy mají omezení, lze na ně volat z jedné IP adresy minimálně s 2-vteřinovými rozestupy.

Endpointy pro testování jsou:

1. GET <http://tasks-rad.quadient.com:8080/task>

Získání zadání, které vrací JSON s unikátním identifikátorem, definicí mapy a 2 body mezi kterými hledáme cestu (jde o klasické X-Y souřadnice, kde **x** určuje pozici na horizontální ose neboli sloupec a **y** na ose vertikální neboli řádek). Mapa je definována jako pole oblastí, přičemž každá oblast je řetězec ve formátu "<obtížnost>-<přechody>", kde přechody mohou nabývat až 4 hodnot ([L]eft, [R]ight, [U]p, [D]own). Pro náš příklad s mapou 3x3 může odpověď vypadat např. následovně:

```
{
  "id": "2727",
  "startedTimestamp": 1503929807498,
  "map": {
    "areas": ["5-R", "1-RDL", "10-DL", "2-RD", "1-UL", "1-UD",
              "2-RU", "1-RL", "2-UL"]
  },
  "astroants": {"x": 1, "y": 0 },
  "sugar": { "x": 2, "y": 1 }
}
```

2. PUT <http://tasks-rad.quadient.com:8080/task/{id}>

(např. pro předchozí zadání: <http://tasks-rad.quadient.com:8080/task/2727>)

Ověření řešení, které přijímá JSON s řešením. Řešení je opět jednoduchý JSON s definicí cesty (1 string), kde každý přechod je opět reprezentován jedním písmenem ([L]eft, [R]ight, [U]p, [D]own). Pro náš příklad s mapou 3x3 může požadavek vypadat např. následovně:

```
{
  "path": "DLDRRU"
}
```

Endpoint na ověření vrací, zda zadaná cesta je validní a zda čas, který uplynul od získání zadání v součtu s náročností průchodu je menší než požadovanou 1 minutu. Toto ověření lze uskutečnit pouze do 5 minut od prvotního získání zadání (poté je třeba vyžádat nové zadání s novým identifikátorem). Odpověď může vypadat např. následovně:

```
{
  "valid": false,
  "inTime": false,
  "message": "You used connection that is not in map from [3,2] going DOWN"
}
```

Vaše řešení posílejte jako ZIP archiv **pouze se zdrojovými kódy** (můžete využít libovolného programovacího jazyka) a případně i s textovým popisem, jak daná aplikace funguje a jak jste úlohu řešili. Lze také poslat odkaz na GitHub, GitLab či libovolné úložiště.

Pokud se Vám nepodaří algoritmus naimplementovat nebo nebudete mít integraci s testovacím API, své řešení nám pošlete také. Rádi se na něj podíváme a bude-li se nám líbit, tak se může taktéž dostat do slosování o ceny. Pravidla slosování a vyhodnocení soutěže naleznete na předchozí stránce.