Zadania krajského kola kategórií A,B

Všeobecné pokyny

Úlohu riešte v jazyku Pascal (dialekt Turbo Pascal v. 7.0 alebo kompatibilný) alebo C (dialekt Turbo C++ v. 3.1 alebo kompatibilný). Zdrojový text vášho riešenia uložte do súboru `MESTO.PAS` resp. `MESTO.CPP` (celé riešenie musí byť uložené v tomto súbore). Môžete použiť štandardné knižnice uvedených dialektov.

Váš program bude čítať vstupné údaje z textového súboru `MESTO.IN` a vypisovať riešenia na obrazovku počítača vo formáte presne stanovenom zadaním úlohy. Môžete predpokladať, že vstupné údaje zodpovedajú popisu v zadaní. Riešenie píšte bielou farbou na obrazovku v štandardnom textovom móde, pred začatím vlastnej činnosti programu nezabudnite zmazať obrazovku, nevypisujte na obrazovku žiadne iné informácie okrem tých, ktoré sú požadované zadaním.

Nemusíte riešiť všetky podúlohy zadania, každá časť je bodovaná samostatne. Programy, ktoré nemožno skompilovať alebo spustiť, nebudú hodnotené.

Zadanie úlohy

Ako iste všetci viete, také typické slovenské mestečko je veľmi zložitá spleť uličiek, budov a trávnikov v ktorej by bez mapy každý zablúdil. Mapa nie je nič iné, než matica rozmerov $N \times M$ obsahujúca rôzne symboly vyjadrujúce, čo v danej časti mesta je.

Na prvom riadku vstupného súboru sú čísla N a M ($0 < N, M \leq 100$) oddelené medzerou, kde N je počet riadkov a M je počet stĺpcov. Riadky a stĺpce číslujeme od 1. Na každom z nasledujúcich N riadkov je M znakov – prvkov matice, teda políčok na mape. Znaky môžu byť takéto:

- '`#`' – budova
- '`+`' – cesta
- '`~`' – voda (napríklad jazero, rieka)
- '`.`' – trávnik (alebo iný nevyužitý priestor)

Vo vstupnom súbore nasledujú ďalšie údaje, tie popíšeme pred úlohou, v ktorej ich budeme potrebovať.

a. *(5 bodov)* Do riadku 2 od pozície 2 vypíšte text: "`V meste je: A1 ciest`", do riadku 3 od pozície 14 vypíšte text: "`A2 krizovatiek`", do riadku 4 od pozície 14 vypíšte text: "`A3 slepych ulic`", do riadku 5 od pozície 14 vypíšte text: "`A4 budov`". Kde križovatka je cesta, ktorá susedí s aspoň tromi inými cestami (za susedné považujeme políčka so spoločnou hranou) a slepá ulica je cesta, ktorá susedí s práve jednou inou cestou. Pričom A_1, A_2, A_3 a A_4 sú počty ciest, križovatiek, slepých ulíc a budov.

b. *(7 bodov)* Cez mesto sa rozhodli postaviť dialnicu. Dialnica zaberie niekoľko celých susedných stĺpcov mapy, pričom políčka, cez ktoré pôjde dialnica bude treba prestavať. Na ďalšom riadku vstupného súboru sú kladné celé čísla k, c_b, c_c, c_v a c_t ($0 < k \leq M$, $0 \leq c_b, c_c, c_v, c_t \leq 1000$) oddelené medzerami, kde k je počet stĺpcov, ktoré dialnica zaberie a c_b, c_c, c_v resp. c_t sú ceny prestavby budovy, cesty, vody resp. trávnika[1].

[1] Domy sa búrajú a cez vodu sa stavajú mosty.

Zistite, za akú najnižšiu cenu sa dá postaviť dialnica a do riadku 7 od pozície 2 vypíšte text: "`Cena dialnice: B`", kde B je táto najnižšia cena.

c. *(8 bodov)* Mesto sa rozrastá a vo veľkom meste musí byť aj veľký futbalový štadión. Futbalový štadión je veľká trávnatá plocha v tvare štvorca. Zistite aký najväčší štadión (čiže štvorec s najväčšou možnou plochou, v ktorom každé políčko je trávnik) možno postaviť. Do riadku 9 od pozície 2 vypíšte text: "`Najvacsi stadion ma plochu: C`", kde C je počet políčok, ktoré obsahuje najväčší štadión.

d. *(10 bodov)* Vo veľkých mestách majú mestskú hromadnú dopravu. V týchto dňoch zavádzajú novú autobusovú linku. Na ďalšom riadku vstupného súboru je číslo z ($0 < z \leq 4000$) – počet zastávok, cez ktoré autobus pôjde. Na nasledujúcich z riadkoch sú popisy jednotlivých zastávok. Popis zastávky je dvojica čísel r a s oddelených medzerou udávajúca riadok a stĺpec, kde sa zastávka nachádza. Autobus prechádza jednotlivé zastávky v poradí, v akom sú uvedené vo vstupnom súbore, pričom začína v prvej a končí v poslednej zastávke. Za desať sekúnd prejde z políčka, na ktorom je na niektoré zo susedných políčok. Na každej zastávke stojí jednu minútu. Zistite za aký čas prejde autobus celú trasu. Do riadku 11 od pozície 2 vypíšte text: "`Prejdenie trasy trva: D sekund`", kde D je najmenší čas, za ktorý autobus prejde celú trasu, pričom chodí iba po ceste. Ak sa daná trasa prejsť nedá do riadku 11 od pozície 2 vypíšte text: "`Trasa sa autobusom prejst neda`".

e. *(20 bodov)* Ráno chodieva veľa ľudí na autách do práce. Firma *47. Programatorska a.s.* začala veľkú reklamnú kanpaň a rozhodla sa rozmiestniť billboardy tak, aby si každý človek ráno po ceste do roboty aspoň jeden z nich všimol. Ale postavenie billboardu je veľmi drahé, a tak by ich chceli postaviť čo najmenej. Na ďalšom riadku vstupného súboru sú čísla d a f – počet budov, kde ľudia bývajú (domy) a počet budov, kde pracujú (firmy). Nasleduje d riadkov s popismi domov a potom ďalších f riadkov s popismi firiem. Popis budovy je dvojica čísel r a s oddelených medzerou udávajúcich riadok a stĺpec, kde tá budova je. Môžete predpokladať, že na uvedených políčkach sú naozaj budovy a každá budova je uvedená najviac raz. Človek cestu z domu do firmy začína na políčku, ktoré je cesta a susedí s domom a končí na políčku ktoré je tiež cesta a susedí s firmou, pričom stále ide po ceste. Ľudia môžu ísť medzi ľubovoľnou dvojicou domu a firmy, medzi ktorými sa dá popísaným spôsobom prejsť. Uvedomte si, že nemusia (ale môžu) ísť najkratšou cestou. Človek uvidí billboard len vtedy, keď prechádza cez políčko, na ktorom je billboard postavený. Do riadku 13 od pozície 2 vypíšte text: "`Postavit terba: E billboardov`", kde E je najmenší možný počet billboardov, ktoré treba postaviť.

f. *(20 bodov)* Cesty v meste sú príliš úzke na to, aby po nich chodili autá obidvomi smermi. Preto treba niektoré cesty zjednosmeriniť. Dvojica susedných políčok, na ktorých je cesta sa zjednosmerní tak, že dovolíme, aby autá medzi nimi chodili len jedným smerom. (Teda z prvého na druhé, ale nie z druhého na prvé.) Iné dvojice políčok sa zjednosmeriniť nedajú. Zistite, koľko najviac rôznych dvojíc políčok môžeme zjednosmerniť tak, aby sa potom dalo prejsť medzi každými dvomi miestami, medzi ktorými sa dá prejsť aj bez zjednosmernenia. Do riadku 15 od pozície 2 vypíšte text: "`Zjednosmernit mozno: F ciest`", kde F je počet, ktorý ste zistili.

g. *(15 bodov)* Veľká organizovaná skupina turistov ide na výlet autobusom. Na túto cestu si musia pre všetkých kúpiť cestovné lístky. Lístky sú rôznych typov. Na ďalšom riadku vstupného súboru je číslo p ($0 < p \leq 100$) – počet typov lístkov. Na ďalších p

riadkoch sú popisy jednotlivých typov lístkov. Popis typu je štvorica čísel s, d, l a c ($0 \leq s, d, l \leq 5$, $0 \leq c \leq 100$) oddelených medzerou. Kde s, d a l sú počty študentov, detí a ostatných (dospelých) ľudí, ktorí môžu (naraz, ale nie nutne všetci) cestovať na jeden lístok toho typu a c je cena jedného lístka toho typu. Môžete predpokladať, každé z čísel s, d a l je aspoň pri jednom lístku nenulové. Na ďalšom riadku vstupného súboru je trojica čísel t_s, t_d a t_l ($0 \leq t_s, t_d, t_l \leq 50$) – počet študentov, detí a ostatných ľudí v skupine turistov. Zistite, koľko najmenej musia za lístky zaplatiť, aby sa mohli všetci odviezť. Do riadku 17 od pozície 2 vypíšte text: "`Turisti zaplatia: G`", kde G je cena, ktorú musia zaplatiť.

h. *(15 bodov)* Jeden turista zo skupinky zmeškal autobus, ale našťastie má malú mapku miesta na ktoré šli. Zistite na koľko miest mohli ísť. Na nasledujúcom riadku vstupného súboru sú čísla n a m ($0 < n \leq N$, $0 < m \leq M$) oddelené medzerou udávajúce počet riadkov a stĺpcov tejto mapky. Na nasledujúcich n riadkoch je popis mapky v takom istom formáte ako popis mapy celého mesta[2]. Do riadku 19 od pozície 2 vypíšte text: "`Mohli ist na: H miest`", kde H je počet rôznych obdĺžnikov s n riadkami a m stĺpcami (prípadne prekrývajúcich sa) ma mape mesta ktoré sú zhodné s touto malou mapou.

[2] Až na to, že v riadku je m, nie M znakov

Príklad vstupu:
(Vstup je dlhší, a preto je tu rozdelený do stĺpcov, vo vstupnom súbore je to pod sebou.)

```
10 30
...~~~..##################....
++++++++++++++++++++++++++++++
...~~~###+++#####+++#########+#
...~~..#++##########++#.....#+#
##.~~.#++#.#+++++++++#.....#+#
##~~..#+#..#+~~#####+#.....#+#
..~~..#+#..#+~~....#+#.....#+#
.~~...#++##.#....##++#.....++#
~~..+..#+++#######+++#.......+.
~.......##+++++++++##..........
5 900 0 1000 100
5
2 3
5 13
10 11
9 29
8 28
5 4
3 15
4 14
5 2
8 30
10 9
6 2
3 16
3 17
9 13
4
1 0 0 10
0 1 0 10
0 0 1 20
1 1 1 25
0 5 1
2 5
..#+#
..#+#
```

Príklad výstupu:
(znaky '~' reprezentujú medzery; takto by mala vyzerať obrazovka po spracovaní uvedeného vstupu):

```
~~~
```