Zadania celoštátneho kola kategórií A,B

Všeobecné pokyny

Tvojou úlohou bude napísať DVA programy. Prvý z nich (teória) bude rovnakého typu ako riešenie krajského kola. Druhý (prax) bude netradičnejší. Je na tebe, ktorému z nich venuješ koľko času. Ak chceš, môžeš dokonca odovzdať riešenie iba jednej časti. Body za riešenia oboch častí sa ti sčítajú.

Úlohu rieš v jazyku Pascal (dialekt Turbo Pascal v. 7.0 alebo kompatibilný) alebo C (dialekt Turbo C++ v. 3.0 alebo kompatibilný). Môžeš použiť štandardné knižnice uvedených dialektov. Programy, ktoré nemožno skompilovať alebo spustiť, dostanú 0 bodov.

Zadanie úlohy – teória (150 bodov)

Riešenie tejto časti by malo byť celé uložené v súbore `TEORIA.PAS/.CPP`. Tvoj program bude čítať vstupné údaje z textového súboru `TEORIA.IN` a vypisovať riešenia na obrazovku počítača vo formáte presne stanovenom zadaním úlohy. Môžeš predpokladať, že vstupné údaje zodpovedajú popisu v zadaní. Riešenie píš bielou farbou na obrazovku v štandardnom textovom móde. Pred začatím vlastnej činnosti programu nezabudni zmazať obrazovku. Nevypisuj na obrazovku žiadne iné informácie okrem tých, ktoré sú požadované zadaním. **Nemusíš riešiť všetky podúlohy zadania, každá časť je bodovaná samostatne.**

Už od detstva bolo tvojím snom byť hrdinom. Nosiť meč, štít a brnenie, strieľať z luku, vedieť kúzliť, neohrozene bojovať proti drakom, škretom či vlkodlakom a zachraňovať krásne dámy[1] zo smrteľného nebezpečenstva. Preto keď si vyrástol, neváhal si ani okamih. Rozhodol si sa, že si zbalíš svojích "pět švestek" a pôjdeš do sveta šťastie hľadať. Ale za tie roky zaháľania si sa bohužiaľ stal závislým na počítači, preto si sa rozhodol ešte naposledy ho využiť – pri príprave odchodu do sveta.

Vstupný súbor obsahuje údaje o tebe a o svete. Svet, do ktorého sa chystáš, je obdĺžnikového tvaru a skladá sa z $R \times C$ štvorcových políčok. V prvom riadku vstupného súboru sú tieto dve celé čísla (pričom $1 \leq R \leq 23$, $1 \leq C \leq 80$) oddelené medzerou. Riadky mapy číslujeme od 1 do R a stĺpce od 1 do C.

Nasledujúcich R riadkov vstupného súboru popisuje mapu sveta. V každom z nich je reťazec presne C znakov s ASCII hodnotami 33-127 (t.j. bez medzier). Pritom znak '#' predstavuje políčko, na ktorom nemôžeš stáť, ani cez neho nevidíš. Všetky ostatné znaky predstavujú políčka, na ktorých stáť môžeš. Znak '.' predstavuje prázdne políčko, znak '*' kríky a znak '@' políčko, kde začínaš svoju púť svetom (také je práve jedno, na tomto políčku momentálne kríky nerastú). Pri svojej púti sa môžeš hýbať ôsmimi smermi (t.j. na jeden krok zmeniť každú svoju súradnicu najviac o 1).

a. *(10 bodov)* Do riadkov od 2. a stĺpcov od 40. vykresli ľavý horný roh mapy. Presnejšie prvých 10 riadkov mapy, z každého riadku prvých 30 stĺpcov. Ak má mapa riadkov alebo stĺpcov menej, na príslušné políčka vypíš medzery.

[1] dievčatá: pohľadných mladých mužov

b. *(30 bodov)* Vo svete možno sú aj políčka, na ktorých by si síce mohol stáť, ale nevieš sa na ne dostať z miesta, kde začínaš. Tvoj prastrýko ti kedysi daroval kúzelnú paličku, ktorú keď použiješ na susedné políčko, zmení ho na prázdne a teba kúzlom presunie naň. Presun na voľné susedné políčko ti trvá 1 minútu, použitie paličky 2 dni. Paličku môžeš použiť ľubovoľne veľa krát. Najvzdialenejší kút sveta je to políčko, kam by ti najdlhšie trvalo dostať sa. Ak ich je viac, nájdi to v najmenšom riadku, ak ich je stále viac, tak to v najmenšom stĺpci. Do riadku 2 od pozície 2 vypíš text "`Najvzdialenejsi kut: X Y`", kde X, Y sú nájdené súradnice.

c. *(20 bodov)* Zisti aj najmenší čas potrebný na cestu do najvzdialenejšieho kútu sveta. Do riadku 3 od pozície 2 vypíš text "`Cas na cestu don: D dni, HH:MM`". Pritom počet hodín aj minút vypíš s práve dvomi ciframi (nie 7:3, ale 07:03).

d. *(20 bodov)* V našom svete sa príroda mení ako divá. Kde ešte pred minútou nič nerástlo, je teraz hotová džungľa. Alebo aj nie. Po chvíli pozorovania si ľahko odhalil, akými pravidlami sa príroda riadi. Situácia sa mení každú minútu. Ak na políčku rástli kríky, budú tam naďalej **práve vtedy**, ak rástli kríky aj na práve 2 alebo 3 z okolitých 8 políčok. Ak na políčku kríky nerástli, začnú tam rásť **práve vtedy**, ak rástli kríky na práve 3 okolitých políčkach. Navyše kríky nemôžu rásť na políčku, kde sa nedá stáť.

   Na ďalšom riadku vstupného súboru je jedno celé číslo M ($1 \leq M \leq 1\,000$). Do riadkov od 13. a stĺpcov od 40. vykresli ľavý horný roh mapy po M minútach. Presnejšie prvých 10 riadkov mapy, z každého riadku prvých 30 stĺpcov. Ak má mapa riadkov alebo stĺpcov menej, na príslušné políčka vypíš medzery. Na políčku kde stojíš vypíš '@' bez ohľadu na to, či tam teraz rastú kríky.

e. *(35 bodov)* Navyše by ťa zaujímalo, či sa počas tvojho pobytu v prírode niektoré rozloženie kríkov zopakuje. (T.j. či bude v dvoch rôznych minútach vyzerať mapa presne rovnako.) Zisti, či sa v priebehu prvých M minút niektoré rozloženie kríkov zopakuje.

   Ak áno, nájdi najmenšie $T \leq M$ také, že rozloženie kríkov po T minútach je rovnaké ako niektoré skoršie rozloženie. Do riadku 5 od pozície 2 vypíš text "`Prve opakovanie: T`". Ak nie, do riadku 5 od pozície 2 vypíš text "`Do M minut sa neopakuje.`", pričom M je číslo načítané zo vstupu.

f. *(35 bodov)* Už ostáva len jedno. Zbaliť si batoh. Ale keby si len nemusel brať toľko vecí – mobil, notebook, dve fľaše Coca-Coly, 3 krabice pizze, pilník na nechty, sveter, ... To všetko ani len nezbalíš, tobôž neunesieš. Prišiel čas kompromisov.

   V ďalšom riadku vstupného súboru sú dve celé čísla M a N oddelené medzerou ($1 \leq M \leq 1\,000$, $1 \leq N \leq 100$). Pritom M predstavuje tvoju nosnosť v gramoch, N je počet typov vecí, ktoré by si chcel vziať. Nasleduje N riadkov, každý popisuje jeden typ vecí – obsahuje tri celé čísla p_i, m_i, c_i ($1 \leq p_i \leq 1\,000\,000$, $10 \leq m_i \leq 20\,000$, $-47 \leq c_i \leq 10\,000$). Príslušná vec má hmotnosť m_i, užitočnosť c_i a máš k dispozícii p_i kusov tejto veci.

   Zisti, aký najužitočnejší batoh si môžeš zbaliť tak, aby si ho uniesol. Užitočnosť batohu je samozrejme súčet užitočností zbalených vecí. (Ak napr. zbalíš 3 zápalky, do užitočnosti batohu sa ti zaráta 3× užitočnosť zápalky.) Do riadku 7 od pozície 2 vypíš text "`Najlepsie zbaleny batoh: B`", kde B je najväčšia dosiahnuteľná užitočnosť batohu.

Príklad vstupu:

```
21 65
................*.........*...........##.............#..###......
.@....*...*....*.*.....*.*.**..........##..........##...####.....
.....*..**....*...*.....*...*...........#..######.#....##..##....
......**.......*.......*.*.**...........#.##.*.#########...######
.....*.*.......***......*.*.............###.***.##.........######
......**.........*.....*................#...###..#..***....######
.....*..**....*...*.....*...............##.##.##.#.........######
......*...*....*.*.....*.................###...###.........######
................*...............................#.....***##.....
.................................................#......##......
.................................................######........
.......**...............................***...............*......
.......**.................................................*.....
.....................................*******..........***..*....
..***.*****..............................................*....
***...**.............................***********.............*....
..**..*..*.......*.......*..............................*...*....
.......*...*....***.....***........**************......***.......
.....................................................**....**....
.....***...............................................***.......
.......................................................*........
500
100 5
5 17 13
2 35 32
1 38 49
10 10 3
3 256 1500
```

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

```
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
~Najvzdialenejsi~kut:~5~65~~~~~~~~~~~~~.................*.........*...~~~~~~~~~~~
~Cas~na~cestu~don:~6~dni,~01:00~~~~~~~~.@....*...*....*.*.....*.*.**.~~~~~~~~~~~
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~.....*..**....*...*.....*...*.~~~~~~~~~~~
~Prve~opakovanie:~140~~~~~~~~~~~~~~~~~~.....**.......*.......*.*.**.~~~~~~~~~~~
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~.....*.*.......***......*.*...~~~~~~~~~~~
~Najlepsie~zbaleny~batoh:~97~~~~~~~~~~~.....**.........*.....*......~~~~~~~~~~~
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~.....*..**....*...*.....*.....~~~~~~~~~~~
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~.....*...*....*.*.....*......~~~~~~~~~~~
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~................*............~~~~~~~~~~~
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~.............................~~~~~~~~~~~
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~.............................~~~~~~~~~~~
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~.@...........................~~~~~~~~~~~
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~..............**.............~~~~~~~~~~~
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~..............**.............~~~~~~~~~~~
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~.............................~~~~~~~~~~~
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~.......................**....~~~~~~~~~~~
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~......................*..*...~~~~~~~~~~~
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~.......................**....~~~~~~~~~~~
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~.............................~~~~~~~~~~~
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~....................**........~~~~~~~~~~~
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
```

Zadanie úlohy – prax (150 bodov)

Riešenie tejto časti by sa malo skladať z programu uloženého v súbore `PRAX.PAS/.CPP` a mapy, uloženej v súbore `PRAX.MAP`. **Začať riešiť túto časť odporúčame až vtedy, keď už máš hotového dosť veľa z predchádzajúcej časti.** (Táto časť môže byť omnoho náročnejšia na čas!)

Nakoniec si sa rozhodol, že v svete môže byť síce dobre, ale za počítačom je predsa len najlepšie. A keď ťa zabijú, vždy sa dá loadnuť. Preto si radšej namiesto chodenia do sveta napíšeš dungeon.

Základné podmienky.

Celý program musí bežať v štandardnom textovom móde (80×25, 16 farieb) a byť ovládaný výlučne z klávesnice. Bude fungovať kolovo, t.j. vždy počká na ťah hráča, vykoná ho a následne vykoná ostatné veci, ktoré sa udiali do nasledujúceho ťahu hráča.

Bodovaný bude podľa toho, ktoré z predpísaných vecí sa ti podarí v časovom limite implementovať. Je na tebe, ktoré z nich si vyberieš, nemusí to byť v poradí, v akom sú uvedené v zadaní. *Ak program nebude schopný reagovať na príkazy užívateľa v reálnom čase, prípadne ak nastane počas jeho behu chyba, budú tvojmu programu strhnuté body.*

Čo by mal tvoj program vedieť.

- Načítať zo súboru mapu *[v podobnom formáte ako v časti "teória", môžeš si doň doplniť vlastné vylepšenia]* a vykresliť ju na obrazovku. *(5 bodov)*
- Vedieť pomocou klávesnice pohybovať hráča po mape, samozrejme len po políčkach, na ktorých môže stáť. *(5 bodov)*
- Program by sa mal dať korektne skončiť. *[Napr. stlačením 'Q'.] (2 body)*
- Niekde na obrazovke vypisovať všetky udalosti, ktoré sa odohrajú v hre. *(3 body)*
- Hra by sa mala dať kedykoľvek uložiť, uložená hra nahrať do pamäte. Všetko musí byť po nahratí v rovnakom stave ako pred uložením! *(15 bodov)*
- Hráč by mal mať inventár, na mape sú predmety, ktoré vie zobrať a položiť. *(10 bodov)*
- Viditeľnosť – nezobrazuje sa celá mapa, len to, čo hráčovi nezakrývajú steny. *(15 bodov)*
- Prefíkanejšia viditeľnosť – čo už kedysi videl, ale teraz nevidí, si pamätá. *[Napr. jasne vykreslené to, čo vidí, tmavo to, čo si pamätá.] (10 bodov)*
- Na mape by mohli pobiehať rôzne príšerky... *(15 bodov)*
- ... byť rôzne rýchle, rôzne nebezpečné... *(5 bodov)*
- ... keď si dosť blízko, rozbehnú sa k tebe najkratšou cestou... *(10 bodov)*
- ... pričom ak sa im oplatí ísť šikmo alebo rovno, idú šikmo... *(5 bodov)*
- ... a keď k tebe dobehnú, vedia zabiť oni teba aj ty ich. *[Dobrý spôsob ako hráč z klávesnice zaútočí na príšerku – skúsi vôjsť na políčko, kde príšerka stojí.]* Útok by mal uberať obeti zdravie. Pri nulovom zdraví obeť umiera. *(10 bodov)*
- Hráč by mal vedieť kúzliť. Útočné kúzla by samozrejme mali pôsobiť na príšery aj na neho.
  – fireball (ohnivá guľa polomeru > 1, letiaca niektorým z ôsmych smerov) *(5 bodov)*
  – magic missile (kúzelná strela, letí ďalej a odráža sa od stien) *(5 bodov)*
  – invisibility (neviditeľnosť, po čase musí opäť začať byť viditeľný) *(10 bodov)*
- Na kraji mapy by sa malo dať prejsť na inú mapu. Niektoré mapy by mohli byť náhodne generované, najlepšie niekoľko miestností pospájaných chodbami. *(20 bodov)*