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. 3.0 alebo kompatibilný). Zdrojový text vášho riešenia uložte do súboru `SIET.PAS` resp. `SIET.CPP` (celé riešenie musí byť uložené v tomto súbore). Môžete použiť štandardné knižnice uvedených dialektov.

Váš program bude čítať vstupné údaje z neskôr popísaných súborov 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. Váš program musí zbehnúť plynulo a nesmie čakať na žiadny vstup z klávesnice.

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

Vstupné súbory sú v pracovnom adresári vášho programu, a tak nevypisujte explicitnú cestu k týmto súborom. Tu je zoznam všetkých vstupných súborov:

```
PASSWD.TXT
CONNECT.TXT
LOGIN.TXT
PACKETS.BIN
TREZOR.TXT
```

Zadanie úlohy

Obyvatelia planéty MEZ sa rozhodli postaviť si armádnu počítačovú sieť PARANET. Po dlhých rokoch sa im práca podarila aj vďaka pánom zo spoločnosti STATINF. Vašou úlohou je napísať softvér, ktorý dokáže vyhodnocovať isté črty a vlastnosti novovybudovanej siete, a to hlavne z bezpečnostných hľadísk.

Počítačová sieť pozostáva z počítačov so zanedbateľne malými rozmermi, ktoré sa nachádzajú na neskôr špecifikovaných súradniciach. Niektoré z nich sú spolu spojené priamym optickým káblom, ktorý si možno predstaviť ako úsečku spájajucu dva body zodpovedajúce počítačom.

Každý z týchto počítačov má priradeného práve jedného užívateľa a naopak.

a. V pracovnom adresári sa nachádza súbor s heslami užívateľov nazvaný `PASSWD.TXT`
   Formát súboru: Niekoľko riadkov (menej ako 200), pričom každý riadok obsahuje informáciu o istom užívateľovi. Konkrétne, pozostáva z jeho mena, medzerou oddeleného zakódovaného hesla, a súradníc počítača patriaceho tomuto užívateľovi. Spôsob zakódovania hesla je uvedený v Dodatku 1. Súradnice každého počítača sú zhodou okolností celé čísla z intervalu -1000 až 1000. Nijaké dva rôzne počítače nemajú rovnaké súradnice.

   Poznamenajme, že koniec súboru sa dá detekovať funkciami *eof*, resp. *feof*.

   Príklad súboru PASSWD.TXT:

```
jano fg67kl 10 20
jozo 56fz651 1 -4
imrich faklio7 -8 10
```

Môžete predpokladať, že každé meno je max. 12 znakov dlhé a heslo max. 16 písmen dlhé, pričom sa skladajú len s alfanumerických znakov(t.j. malých a veľkých písmen anglickej abecedy a 10 číslic). Ďalej predpokladáme, že neexistujú dvaja užívatelia s rovnakým menom.

V pracovnom adresári sa ďalej nachádza súbor `CONNECT.TXT`, ktorý obsahuje niekoľko riadkov (menej ako 4000). Na každom riadku sú 2 mená oddelené medzerou, ktoré znamenajú, že počítač užívateľa s prvým menom je spojený priamym optickým káblom (úsečkou) s počítačom užívateľa s druhým menom.

Príklad súboru `CONNECT.TXT`:

```
jano jozo
jozo imrich
```

Ak sa však napríklad užívateľ Jano spojí s Jožom a neskôr sa chce Jožo spojiť s Janom, nevytvára sa už nové spojenie, ale sa využije už existujúce.

Vašou úlohou je zistiť dĺžku optického káblu potrebného na uskutočnenie všetkých týchto prepojení. Váš program musí **na prvý riadok od pozície 1** vypísať text `Dlzka␣optickeho␣kabla:␣` nasledovaný dĺžkou potrebného optického káblu. Dĺžku vypíšte zaokruhlenú na dve desatinné miesta, pričom medzi prvou číslicou vypísaného čísla a slovom `kabla` musí byť práve jedna medzera, ako je vyššie uvedené. Toto pravidlo platí aj pre ďalšie úlohy, ktoré vyžadujú výstup reálneho čísla.

b. Tak, ako sú počítače navzájom poprepájané, vytvárajú isté pracovné skupiny. Pracovná skupina je čo najväčšia možina užívateľov, z ktorých je každý s každým spojený (nie priamo, ale trebárs cez ďalších 10 počítačov).

   Presnejšie povedané, užívateľ A **je spojený** s užívateľom B, ak

   - existuje medzi A a B **priame spojenie** (A sa spojil s B alebo B sa spojil s A)
   - existuje užívateľ C taký, že A je **priamo spojený** s C a C **je spojený** s B.
   - inak nie je A spojený s B

   Váš program vypíše **na riadok 2 od pozície 1** text `Pocet␣pracovnych␣skupin:␣` nasledovaný počtom pracovných skupín, ktoré vznikli. Ďalej vypíše **na riadok 3 od pozície 1** text `Najvacsi␣pocet␣clenov␣skupiny:␣` nasledovaný počtom členov najväčšej pracovnej skupiny.

c. V pracovnom adresári sa ďalej nachádza súbor `LOGIN.TXT`.
   Formát súboru: Obsahuje niekoľko riadkov. Na každom riadku je zaznamenaný pokus o prihlásenie sa užívateľa na svoj počítač alebo odhlásenie sa užívateľa. Konkrétne každý riadok pozostáva zo slova `IN` alebo `OUT`, mena užívateľa a nezašifrovaného hesla.

   Pokiaľ užívateľ s týmto menom naozaj existuje (je uvedený v súbore `PASSWD.TXT`) a heslo odpovedá heslu tohto užívateľa, operácia sa uskutoční. Avšak, odhlásenie sa (out) nemá účinok, ak nebol prihlásený(in). Viacnásobné prihlásenie tiež nemá žiadny účinok, užívateľovi neumožníme uskutočniť prihlásenie, ak už prihlásený je.

   Príklad:

```
in brano Jankasicool
out dusan 7Matkajesuper
out brano Jankasicool
in martin love8zuzka
in martin love8zuzka
```

Táto postupnosť spôsobí, že sa Braňo prihlási aj odhlási, Dušanovi sa odhlásenie nepovolí a Martin sa prihlási, no neodhlási.

Zistite, koľkí užívatelia zostanú na konci takejto postupnosti operácií prihlásení (v našom prípade 1, a to Martin).

Váš program **na riadok 4 od pozície 1** vypíše text `Pocet␣prihlasenych:␣` nasledovaný počtom užívateľov, ktorí zostali prihlásení.

d. Keďže si po našej sieti budeme posielať rôzne súbory, budeme potrebovať skontrolovať, či všetky dáta prišli korektne. Navyše, aby sme nemuseli posielať znovu celý súbor, ak sa dáta niekde poškodili, budeme jednotlivé súbory deliť na malé „pakety", ktoré budú obsahovať potrebné informácie na znovuzostavenie pôvodného súboru a taktiež informácie použité na jednoduchú kontrolu prijatých dát. Popis „paketu" nájdete v Dodatku 2.

   Vašou úlohou je napísať funkciu, ktorá z paketov, ktoré prišli na náš pocítač, vyberie správu s názvom `Paranet@` určenú pre užívateľa menom `Petrik`. Túto správu potom vypíšte **od začiatku riadku 6**. Všetky pakety načítajte z binárneho súboru `PACKETS.BIN`.

   Nezabudnite, že každý paket ma pevnú dĺžku 36 bajtov. Ak sa vyskytol paket so zlým kontrolným číslom (či už kontrolným bajtom hlavičky, alebo kontrolným bajtom dát) nesmiete tento paket použiť. Správu poskladáte z korektných paketov, a ak bude nakoniec nejaký paket chýbať, nahradíte celé jeho možné dáta jedným znakom "#".

   Môžete predpokladať, že každá správa má dĺžku najviac 100 znakov a každá správa je rozsekaná najviac na 100 paketov. Ak má správa viac než 80 znakov, vypíšte ju obyčajným spôsobom na obrazovku, aj keď bude text zle zalomený.

e. Ďalším ohrozujúcim faktorom počítačovej siete PARANET je jadrový reaktor v blízkosti. Užívatelia tejto siete sa rozhodli zabezpečiť proti prípadnej katastrofe a plánujú oplotiť svoje pracovisko platinovou stenou. Keďže ale tento kov je vééééééľmi drahý, musia šetriť a navrhnúť pôdorys tak, aby mal čo najmenší obvod. Po dlhom premýšľaní sa im podarilo prísť na to, že to musí byť konvexný obal okolo počítačov. Vašou úlohou je zistiť dĺžku tohto konvexného obalu a **na riadok 8 na pozíciu 1** vypíšte text
   `Obvod␣platinoveho␣obalu:␣` nasledovaný obvodom zaokrúhleným na 2 desatinné miesta.

f. Pri zadnej stene (rovnobežná s osou x) počítačovej miestnosti stojí trezor, ktorý sa rozhodla miestna banka odkúpiť. Keďže všetky stoly s počítačmi sú pripevnené k podlahe skrutkami, zaujímame sa, či trezor vzhľadom na svoje rozmery `Tx, Ty` dokážeme posúvaním premiestniť pomedzi počítače na druhú stranu miestnosti k východu. Trezor sa dá posúvať medzi počítačmi, ak je rozdiel ich súradníc x aj y väčší alebo rovný x-ovej, resp. y-ovej veľkosti trezora. Trezor nie je možné otáčať.

Vstupný súbor `TREZOR.TXT` obsahuje dva riadky. Na prvom sú štyri čísla definujúce ľavý horný roh (najskôr x-ová, potom y-ová súradnica) a pravý doľný roh obdĺžnikovej miestnosti. Druhý riadok obsahuje dve čísla definujúce rozmery trezora (opäť najskôr rozmer na osi x).

Vašou úlohou je pre daný vstup určiť, či sa trezor dá posúvaním dostať na druhý koniec miestnosti (k prednej stene). **Rozmery počítačových stolov zanedbávame** a môžete predpokladať, že vzdialenosti prednej a zadnej steny (steny rovnobežné s osou x, pri zadnej stene je trezor) od úrovne najbližších počítačov je väčšia ako y-ový rozmer trezoru (čo znamená, že trezorom môžeme ľubovoľne posúvať vľavo, vpravo, pokiaľ ním nepohneme smerom dopredu od zadnej steny, a taktiež vľavo, vpravo, keď sa už dostaneme za úroveň posledného počítača pri prednej stene).

Na riadok 9 od pozície 1 potom vypíšte text
`Trezor␣sa␣da␣presunut`
resp. text
`Trezor␣sa␣neda␣presunut`

g. Pokrok nezastavíš. Správca siete v spolupráci s firmou STATINF zabezpečili inováciu siete. Inovácia znamená, že niektoré priame spojenia budú nehradené novšími, infrafialovými spojeniami s vysokou rýchlosťou. Podmienkou je, aby tieto zachovávali pracovné skupiny. To znamená, že ak dva počítače patrili do istej pracovnej skupiny, tak medzi nimi musí existovať nejaké infrafialové spojenie (nie nevyhnutne priame).

   Nájdite dvojice počítačov, medzi ktorými je nutné natiahnuť infrafialovú linku tak, aby pracovné skupiny zostali zachované, no súčet dĺžok infrafialových spojení bol minimálny.

   Váš program by mal **na riadok 10 od pozície 1** vypísať text `Minimalna␣dlzka␣infrafialoveho␣spojenia:␣` nasledovaný onou minimálnou dĺžkou zaokrúhlenou na dve deastinné miesta.

Dodatok 1

Kód hesla získame v dvoch krokoch:

1. Najprv v zadanom hesle urobíme cyklický posun doprava o 7 znakov na každom písmene hesla, pričom pracujeme na abecede tvaru (0,1,...,9,A,B,...,Z,a,b,...z) - čo znamená, že napr. 0 sa zobrazí na 7, 4 na B, F na M, Z na g, s na z, a nakoniec cyklicky napr. x na 4, z na 6.

2. Jednotlivé, takto prepísané písmenka, teraz presunieme na inú pozíciu podľa danej mriežky a všetky vzniknuté voľné miesta vynecháme.
   Mriežka : 4 9 14 11 5 3 15 1 16 12 7 8 10 2 13 6

   Napríklad zo slova `ETa12I5opr9` by po prechode touto mriežkou vzniklo slovo `o␣IE2␣9␣T␣1r␣a5p`, a po vynechaní voľných miest nakoniec dostaneme `oIE29T1ra5p`.

Dodatok 2

V súbore `PACKETS.BIN` sa nachádzajú jednotlivé pakety. Dĺžka každého paketu bude pevná, a to 36 bajtov.

Štruktúra paketu:

Head length (1 byte) dĺžka hlavičky paketu = počet bajtov, ktoré zaberá hlavička paketu. (v každom pakete sa môžu vyskytovať aj rôzne dodatočné prepínače). Za hlavičkou sa hneď nachádzajú prenášané dáta z rozsekaného súboru

Position number (1 byte) poradové číslo paketu vzniknutého rozsekaním daného súboru (1..number of packets)

Number of packets (1 byte) počet všetkých paketov vzniknutých rozsekaním daného súboru

Source name (presne 8 bytes) názov rozsekaného súboru

Destination (niekoľko, maximálne však 12 znakov (char), ukončených nulou) meno počítača, pre ktorý je tento paket určený (pre rôzne počítače možu byť vytvorené rôzne pakety z toho istého súboru)

Header checksum (1 byte) kontrolný súčet hlavičky - ten sa vytvorí postupným XOR-ovaním všetkých, **doteraz** použitých bajtov (napr. (Head length) XOR (Position number) XOR ...), samozrejme okrem Header checksum.

Options tu je niekoľko dodatočných prepínačov, ktoré sú až do konca hlavičky

(tu je koniec hlavičky paketu)

Data dátová časť - prenášané dáta rozsekaného súboru

Data checksum (1 byte) kontrolný súčet dát - vytvorí sa postupným XOR-ovaním všetkých bajtov v dátovej časti

0.1 Príklad vstupu a výstupu

- Súbor PASSWD.TXT:

```
Adam zvhu5 10 1
Joseph 0oy01 15 16
George jykjlvth5 20 20
Jim vjhlysjlpvw 15 10
William kztl0yyuwpl 4 20
Petrik pzwjyvv 4 14
```

- Súbor LOGIN.TXT:

```
IN William mrpresident
IN David scorpius
IN Joseph truth
IN Petrik scorpio
IN George democracy
IN Jim Alice
IN Adam sayno
IN Jim alicecooper
```

- Súbor CONNECT.TXT:

```
Adam Jim
Petrik William
Joseph Seorgius
Joseph George
George Petrik
Petrik Jim
Adam Petrik
Joseph Petrik
```

- Súbor TREZOR.TXT:

```
0 30 22 -10
8 5
```

- Výstup:

```
Dlzka optickeho kabla: 76.99
Pocet pracovnych skupin: 1
Najvacsi pocet clenov skupiny: 6
Pocet prihlasenych: 6

0123456789012345678901234567890

Obvod platinoveho obalu: 57.79
Trezor sa da presunut
Minimalna dlzka infrafialoveho spojenia: 45.58
```