Zadania Celoslovenského kola súťaže ZENIT v programovaní v školskom roku 2006/2007 pre kategórie A, B.

Všeobecné pokyny.

Čas na riešenie úloh je **4 hodiny 30 minút.**

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 **priezvisko.PAS** resp. **priezvisko.CPP** (celé riešenie musí byť uložené v tomto súbore). Ak robíte **v jazyku C/C++, vytvorte aj EXE súbor**. Môžete použiť štandardné knižnice uvedených dialektov. **Nezabudnite do komentárov na začiatku programu uviesť svoje meno, triedu a školu.**

Váš program bude čítať vstupné údaje zo súboru **VSTUP.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. **Pri nejasnostiach sa riaďte ukážkou výstupu alebo testovacím programom.** Ostré bodovanie prebehne na celkom iných vstupných súboroch. Pri určovaní poradia môžu zohrať rozhodujúcu úlohu prémiové body za kratší čas behu vášho programu pri ostrých testoch.

Zadanie úlohy: Ujo Fero a zaplátaná strecha.

Napíšte program (a **uložte ho pod svojím menom v komentári na úvod uveďte svoje meno, triedu a školu**), ktorý vyrieši tieto problémy:

0)(12 bodov) Od pozície (2,2) vypíše text: **'Zaplaty na streche.'**.

a) (3+5+4+4 bodov) Prečíta zo vstupného súboru údaje o záplatách na streche a na pozíciu (2,4) vypíše ich počet. Na pozíciu (2,5) vypíše celkovú plochu záplat. Na pozície (2,6-7) vypíše plochu najväčšej a najmenšej z nich.

b) (5+5+5+6+5+8 bodov) Ujo Fero sa vytešuje keď sa dve záplaty prekrývajú a hovorí: **'držia dupľom'**. O počte možných dvojíc nám hovoria kombinačné čísla. Na pozíciu (2,9) napíše text: **'Dvojice: '** a ich počet. Na pozície (2,10-13) vypisuje počty tých, ktoré sa prekrývajú úplne, čiastočne, iba na obvode (ale úsečkou a nie iba bodom) alebo vôbec nie. **Dá sa počítať aj percento prekrytia** dvojice vytešuje sa ujo Fero a to tak, že vypočítam percento spoločnej plochy vzhľadom na plochu, ktorú pokryjú tieto dve záplaty spolu. Najväčšie percento prekrytia vypíše na (2,14).

c) (13+15+5 bodov) Fero uvažuje ako sa tie dvojice záplat prekrývajú, a tak vytvárajú väčšie súvislé celky. Na pozíciu (2,16) vypíšte, koľko ich je. Keby sme použil špeciálnu **LepiPásku** na záplaty, ktorú nám ponuka firma **BestTape** mohli by sme počet týchto častí zredukovať (zlepili by som tie časti, ktoré majú spoločnú úsečku na obvode). Na pozície (2,17-8) napíšte, či sa oplatí nakúpiť takúto pásku a kontaktovať túto firmu. Ak áno, vypíšte o koľko sa zredukuje počet súvislých celkov.

d) (15+20+15 bodov) Najlepší matematik triedy Viktor chce pomôcť ujovi Ferovi a preto hovorí: „Keď sa to zoberie čisto geometricky, strany obdĺžnikov rozdelia celú strechu na veľa menších častí“ (menej ako 2700). Ujo Fero sa teší a hneď tu má matematicky presný odhad:

„to je fantastické teoreticky by to mohlo byť najviac $\mathbf{2}^N$ častí, každá časť je pokrytá inou množinou záplat (? nemusí ak nesusedia) a ich počet môže byť najviac **N“**. Urobte Viktorovi a najmä ujovi Ferovi radosť a vypočítajte, na koľko častí sa rozpadnú záplaty podľa Viktorovho geometrického delenia. Potom je už iba hračkou zistiť aká je plocha najväčšej časti a koľko záplat je najviac na sebe. Výsledky vypíšte od pozície (25,5-7).

e) (3+10+15+15+10+10 bodov) Logika geometrie nepopustí - to sa nám teda rozpadne aj strecha na ktorej nie sú záplaty. Ďalej ujo Fero potrebuje kontrolu, či plocha pokrytá záplatami a nepokrytá strecha dávajú spolu naozaj plochu celej strechy – obdĺžnika (to je ako skúška správnosti! vytešuje sa ujo Fero a v zamyslení si prečesáva šedivú brahu). Potrebnú bilanciu vypíšte na pozície (25,9-14).

Ukážka toho, čo by mal obsahovať Váš program:

```
uses uTest, crt;
var  N,i, MaxX, MaxY : longint;
begin
  clrscr;
  gotoxy(2,2);  write( 'Zaplat ..');
  assign(input,'VSTUP.in');  reset(input);
  readln(N, MaxX,MaxY);
  for i := 1 to N do begin
       ... tvoj program - čítanie ďalších dát
end.
```

```
Vstupný súbor 1: VSTUP.IN
7  20 15        N < 56   maxX, maxY < 999
 9  0  15  2
 2  1  10  6
14  1  17  4
 9  3  16  7
 3  8   5 12
 5  9   9 11
 7 11   9 13
```

Obrazovka testu: 1

```
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
~Zaplaty~na~streche.~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
~Pocet:~7~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
~Plocha:~109~~~~~~~~~~~~Rozpad~na:~11~casti~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
~Max.:~40~~~~~~~~~~~~~~~Najvacsia~cast:~36~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
~Min.:~4~~~~~~~~~~~~~~~~Najviac~vrstiev:~2~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
~Dvojice:~21~~~~~~~~~~~~B~i~l~a~n~c~i~a~:~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
~Uplne:~0~~~~~~~~~~~~~~~Suvisle~nepokryte~kusy:~~~2~~~~~~~~~~~~~~~~~~~~~
~Ciastocne:~4~~~~~~~~~~~Plocha~pokryta~zaplatami:~102~~~~~~~~~~~~~~~~~~~
~Na~obvode:~2~~~~~~~~~~~Plocha~bez~Zaplat:~~~~~~~~198~~~~~~~~~~~~~~~~~~~
~Vobec:~15~~~~~~~~~~~~~~Plocha~celej~strechy:~~~~~300~=~102~+~198~~~~~
~Percento:~5.7~%~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~300~=~20~x~15~~~~~~~
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
~Suvisle:~4~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
~Redukcia~o:~2~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
~BestTape~sa~oplati~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
```

(výpis je iba na šírku 70, medzery sú vypisované ako '~')

Zadania Celoslovenského kola súťaže ZENIT v programovaní v školskom roku 2006/2007 pre kategórie A, B.

Všeobecné pokyny.

Čas na riešenie úloh je **3 hodiny 30 minút.**

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 **priezvisko.PAS** resp. **priezvisko.CPP** (celé riešenie musí byť uložené v tomto súbore). Ak robíte **v jazyku C/C++, vytvorte aj EXE súbor**. Môžete použiť štandardné knižnice uvedených dialektov. **Nezabudnite do komentárov na začiatku programu uviesť svoje meno, triedu a školu.**

Váš program bude čítať vstupné údaje zo súboru **VSTUP.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. **Pri nejasnostiach sa riaďte ukážkou výstupu alebo testovacím programom.** Ostré bodovanie prebehne na celkom iných vstupných súboroch. Pri určovaní poradia môžu zohrať rozhodujúcu úlohu prémiové body za kratší čas behu vášho programu pri ostrých testoch.

Zadanie úlohy: Problémy kartografov.

Vo **Vojenskom kartografickom ústave v Harmanci** spracovali digitálnu mapu regiónu. Región je reprezentovaný mriežkou o rozmeroch **Xmax x Ymax**. Na mriežke je vyznačených **N** usadlostí, pričom každá má celočíselné súradnice **X[i],Y[i]** a ku každej vieme názov **Name[i]** a počet obyvateľov - **O[i]** (**0 < X[i] < Xmax, 0 < Y[i] < Ymax, 0 < O[i] < 1000000**). Vo vstupnom súbore sú v prvom riadku čísla **N**, **Xmax** a **Ymax**. **N** je počet usadlostí. Na ďalších **N** riadkoch sú tri čísla **X,Y,O** a **názov** usadlosti. Potom sú ďalšie riadky s údajmi niektoré ďalšie úlohy. Kartografi rozdeľujú podľa počtu obyvateľov usadlosti do nasledujúcich kategórii:

A 100000 a viac
B 10000 až 99999
C 5000 až 9999
D 1000 až 4999
E 500 až 999
M menej ako 500

Zatiaľ ich digitálna mapa nie je celkom presná. Rozmery mriežky sú najviac **1000 x 1000** a počet usadlostí **N < 125**. Ďalej môžete predpokladať, že neexistujú usadlosti s rovnakými súradnicami.

Napíšte program (a **uložte ho pod svojím menom. V komentári na úvod uveďte svoje meno, triedu a školu**), ktorý vyrieši tieto problémy:

0) (10 bodov) Na pozíciu (2,2) vypíše názov: '**Vojensky kartograficky ustav, Harmanec.**'.

a) (14+12 bodov) Prečíta údaje o usadlostiach zo vstupného súboru '**VSTUP.in**' a do stĺpca od pozície (30,4) vypíše abecedný register usadlostí v tvare podľa ukážky (pre obmedzenie dané veľkosťou obrazovky vypíše iba prvých 44, ak na to je jeden stĺpec nestačí píše aj do druhého od pozície (44,4)). Do stĺpca od pozície (60,2) vypíše zoznam obcí podľa počtu obyvateľov (ak ich je viac ako 23 vypíše iba prvých 23). Ak dve obce majú rovnaký počet obyvateľov, tak ich vypíše podľa abecedy.

b) (7+8 bodov) Na pozíciu (2,4) vypíše počet všetkých obcí a súčet všetkých obyvateľov. Od pozície (2,5) do stĺpca pod sebou vypíše prehľad usadlostí podľa kategórií. Vypíše počet obcí, počet obyvateľov a aj percentuálny podiel na celkovej populácií.

c) (7 bodov) Vo vstupnom súbore na ďalšom riadku prečítame súradnice dvoch bodov **(Ax;Ay) (Bx;By)**. Tieto body tvoria protiľahlé vrcholy obdĺžnika v regióne. Od pozície (2,12) vypíše počet usadlosti v obdĺžniku spolu so súčtom obyvateľov. Usadlosť na hranici patrí do obdĺžnika.

d) (7+15+12 bodov) Na pozíciu (2,14) vypíše najmenšiu vzdialenosť dvoch usadlostí. Podľa nového zákona sa môžu zlúčiť 2 a viac usadlostí do jedného administratívneho celku, ak ich vzdialenosť nie je väčšia ako **V**. Prečíta číslo **V** a na pozíciu (2,15) vypíše, koľko takýchto obcí sa môže zlúčiť a koľko celkov takto vznikne. Ak **vzd(AB)** ≤ **V** a **vzd(BC)** ≤ **V** potom **ABC** sa môžu zlúčiť aj keď **vzd(AC)** > **V**. Na pozíciu (2,16) vypíše maximálny počet, ktorý sa môže zlúčiť do jedného celku aj s počtom ich obyvateľov (ak je viac možností, vypíše väčší počet obyvateľov).

e) (15+8 bodov) Vláda rozhodla, že v regióne bude zriadi rezerváciu štvorcového tvaru. Nájdite na mriežke najväčší štvorec, ktorý vo vnútri neobsahuje žiadnu usadlosť (na hranici môže, dokonca musí!). Na pozície (2,18-19) vypíše veľkosť rezervácie a súradnice jej severozápadného rohu. Ak ich je viac, tak napíše súradnice najsevernejšie a najzápadnejšie (najskôr podľa severu).

f) (9 bodov) Internetový provider **POLUDNÍK s.r.o.** chce zaviesť internet čo najväčšiemu počtu obyvateľov. Pre svoju optickú linku hľadá poludník (**X-ovú** súradnicu), na ktorej žije najviac obyvateľov. Od pozície (2,21) vypíše najzápadnejšiu súradnicu **X**, na ktorej je najviac obyvateľov.

h) (30 bodov) Po tom, ako dala spoločnosť **POLUDNÍK s.r.o.** do prevádzky poludníkovú optickú linku, získala **grant z EU** na výstavbu siete pre všetky obce. Každá obec sa pripojí na buď na poludníkovú linku alebo na najbližšiu obec tak, aby sa prepojili do siete všetky obce a súčet potrebných prepojení bol minimálny. Veľkosť tejto minimálnej siete vypíše na pozíciu (2,23).

Ukážka toho, čo by mal obsahovať Váš program:

```
uses uTest, crt;
       ... tvoj program - čítanie ďalších dát
end.
```

```
Vstupný súbor 1: VSTUP.IN
6 12 10              N  < 111   Xmax,Ymax < 1000
3 3 18880 Puchov     X,Y O, práve 1 medzera, Názov
3 5  1194 Kockovce
3 8  6058 Belusa
5 4   940 Visolaje
2 7   333 Hloza
1 8  2560 Ladce
1 2 6 6                     Ax,Ay Bx,By .. celé
1.8                         V  .. reálne
```

Obrazovka testu: 1

```
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
~Vojensky~kartograficky~ustav,~Harmanec~~~~~~~~~~~~~~~~~~~~Puchov~~~~~~~~~~18882
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~Belusa~~~~~~~~~~~6058
~~~~6~~~29967~~~~~~~~~~~~~~~~Belusa~~~~~~~~~~~~~~~~~~~~~~~~~Ladce~~~~~~~~~~~~2560
~B~~1~~~18882~~63.01%~~~~~~~~Hloza~~~~~~~~~~~~~~~~~~~~~~~~~Kockovce~~~~~~~~~1194
~C~~1~~~~6058~~20.22%~~~~~~~~Kockovce~~~~~~~~~~~~~~~~~~~~~~Visolaje~~~~~~~~~~940
~D~~2~~~~3754~~12.53%~~~~~~~~Ladce~~~~~~~~~~~~~~~~~~~~~~~~~Hloza~~~~~~~~~~~~~333
~E~~1~~~~~940~~~3.14%~~~~~~~~Puchov~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
~M~~1~~~~~333~~~1.11%~~~~~~~~Visolaje~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
~Obce:~3~21016~ob.~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
~Min.vzd.:~1.41~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
~Zluci~sa:~3~do:~1~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
~Obce:~3~~8951~ob.~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
~Rezervacia:~7~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
~na:~(5;0)~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
~Siet:~4.414~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
```

(výpis je na šírku 80, medzery sú vypisované ako '~')