

MATEMATICKÁ OLYMPIÁDA NA STREDNÝCH ŠKOLÁCH

50. ROČNÍK, školský rok 2000/2001

Zadania úloh celoštátneho kola kategórie P

2. súťažný deň

P-III-4

(Formátovanie textu)

Program: FORMAT.PAS/.C/.CPP

Vstup: FORMAT.IN

Výstup: FORMAT.OUT

Do nového textového editora potrebujeme napísať program slúžiaci na formátovanie textu. Editor pracuje v textovom režime s neproporcionálnym písmom. Všetky znaky sú teda rovnako široké a aj medzera má pevnú šírku rovnakú ako ktorýkoľvek iný znak. Taktiež pomocné symboly v texte (ako napríklad interpunkčné znamienka, zátvorky či úvodzovky) sa spracovávajú rovnako ako akékoľvek iné znaky. Editor je pomerne jednoduchý, takže nepripúšťa delenie slov. Program budeme vždy používať len na formátovanie jedného odseku textu.

Pri formátovaní textu budeme za slovo považovať každú súvislú postupnosť nemedzerových znakov, ktorá je na oboch koncoch ohraničená medzerou, prípadne začiatkom alebo koncom riadku. Pomocné symboly v texte sú teda súčasťou tých slov, od ktorých nie sú oddelené medzerou.

Úlohou formátovania textu je vhodné rozloženie slov do jednotlivých riadkov tak, aby bol celý text zarovnaný „do bloku“ (teda k ľavému aj pravému okraju) pri zadanej šírke riadku. Pritom medzery medzi slovami musia byť pokiaľ možno najmenšie a v texte čo najrovnomernejšie rozložené. Tieto všeobecné požiadavky teraz upresníme: Veľkosti medzier medzi slovami v jednom riadku (okrem posledného riadku odseku) sa môžu líšiť najviac o 1, pred prvým a za posledným slovom v riadku nesmie byť medzera. Pokiaľ je v riadku len 1 slovo, je rozloženie medzier ľubovoľné. V poslednom riadku odseku musia byť slová oddelené práve 1 medzerou a pred prvým slovom nesmie byť medzera.

Ak text spĺňa tieto záväzné požiadavky, potom kvalitu sformátovania odseku hodnotíme trestnými bodmi. Ohodnotenie odseku je súčtom ohodnotení jednotlivých riadkov. Ohodnotenie jedného riadku je dané výsledkom funkcie $F(\text{Width}, \text{Chars}, \text{Words}, \text{Last})$, kde Width je šírka stránky (teda počet znakov v riadku sformátovaného textu vrátane všetkých medzier), Chars je počet nemedzerových znakov v riadku, Words je počet slov v riadku a Last hovorí, či sa hodnotí posledný riadok odseku alebo nie.

Ohodnocovacia funkcia F v programovacom jazyku C vyzerá nasledovne:

```
int F(int Width, int Chars, int Words, int Last)
{
    int Spaces = Width - Chars - Words + 1; /* Počet zbytočných medzier */
    int BasePen = LINEPENALTY;             /* Základné trestné body za riadok */

    if (Spaces < 0)                         /* Nevôjde sa text do riadku? */
```

```

    return INFYPEN;
if (Last)                                /* Posledný riadok? */
{
    if (4*(Chars + Words - 1) <= Width) /* Je posledný riadok príliš krátky? */
        BasePen += SMALLLINEPEN;
    return BasePen;
}
if (Words == 1)                          /* Iba jedno slovo v riadku? */
    BasePen += SINGLEWORDPEN;
return Spaces * Spaces + BasePen; /* Ohodnotenie celého riadku */
}

```

V Pascale je zápis funkcie F podobný:

```

function F(Width, Chars, Words: Integer; Last: Boolean): Integer;
var
    Spaces : Integer;      { Počet zbytočných medzier }
    BasePen : Integer;     { Základné trestné body za riadok }
begin
    BasePen := LINEPENALTY;
    Spaces := Width - Chars - Words + 1;
    if Spaces < 0 then      { Nevôjde sa text do riadku? }
        F := INFYPEN
    else if Last then begin { Posledný riadok? }
        if 4*(Chars + Words - 1) <= Width then { Je príliš krátky? }
            Inc(BasePen, SMALLLINEPEN);
        F := BasePen;
    end
    else begin
        if Words = 1 then   { Iba jedno slovo v riadku? }
            Inc(BasePen, SINGLEWORDPEN);
        F := Spaces * Spaces + BasePen; { Ohodnotenie celého riadku }
    end;
end;

```

Hodnoty konštánt sú:

- $LINEPENALTY = 10$
- $SMALLLINEPEN = 5$
- $SINGLEWORDPEN = 20$
- $INFYPEN = 30\,000$

Vašou úlohou je sformátovať odsek textu pri zadanej šírke riadku čo najkvalitnejšie, teda splniť všetky záväzné požiadavky kladené na formátovanie a pritom dosiahnuť čo najnižšie ohodnotenie odseku trestnými bodmi podľa funkcie F .

Vstup:

Vo vstupnom súbore **FORMAT.IN** je na prvom riadku zadaná požadovaná šírka stránky po sformátovaní. V ďalších riadkoch sa nachádza text odseku, určený na sformátovanie. Môžete predpokladať, že žiadny z týchto riadkov nie je dlhší ako 100 znakov, na začiatku ani na konci žiadneho riadku nie je medzera a medzi jednotlivými slovami v riadku je práve 1 medzera. Vstupný súbor vrátane medzier nebude dlhší ako 10 000 znakov.

Výstup:

Do výstupného súboru `FORMAT.OUT` zapíšte text odseku zo vstupu sformátovaný čo najkvalitnejšie podľa vyššie uvedených zásad (teda s najnižšou možnou hodnotou ohodnocovacej funkcie).

Príklad:

`FORMAT.IN`

40

Each section in this document will have the string "<section>" at the right-hand side of the section title. Each subsection will have "<subsection>" at the right-hand side. These strings are meant to make it easier to search through the document.

`FORMAT.OUT`

(jedno z možných riešení; symbol '␣' označuje medzeru)

Each␣section␣in␣this␣document␣will
have␣the␣string␣"<section>"␣at␣the
right-hand␣side␣of␣the␣section␣title.
Each␣subsection␣will␣have␣"<subsection>"
at␣the␣right-hand␣side.␣These␣strings
are␣meant␣to␣make␣it␣easier␣to␣search
through␣the␣document.

P-III-5

(Okružné trasy)

Program: `OKRUZNE.PAS/.C/.CPP`

Vstup: `OKRUZNE.IN`

Výstup: `OKRUZNE.OUT`

V meste Turiststadt začal rozkvetáť turistický ruch. Aby podporili jeho ďalší rozkvet, rozhodla sa mestská rada založiť spoločnosť City-tour. Poslaním tejto spoločnosti je prevádzkovať v meste niekoľko vyhliadkových okružných autobusových liniek. Vašou úlohou je napísať program, ktorý navrhne trasy jednotlivých liniek podľa požiadaviek mestskej rady, prípadne zistí, že sa ich požiadavky nedajú splniť.

Mesto je tvorené križovatkami, ktoré sú navzájom pospájané ulicami. Každá ulica spája práve dve križovatky. Dve križovatky môžu byť spojené viacerými rôznymi ulicami. Križovatkou voláme aj miesto, na ktoré vedú len jedna alebo dve ulice.

Mestská rada kladie na plánované trasy liniek nasledovné požiadavky: Aby si turisti mohli pohodlne prezrieť každú ulicu v meste a pritom neboli prevádzkové náklady príliš veľké, musí každou ulicou prechádzať práve jedna autobusová linka. Žiadna z trás nesmie prejsť niektorou križovatkou viac ako raz. Trasy liniek musia byť okružné, teda prvá a posledná križovatka na trase musia byť rovnaké.

Vstup:

Prvý riadok vstupného súboru **OKRUZNE.IN** obsahuje dve čísla oddelené medzerou – počet križovatiek N ($1 \leq N \leq 120$) a počet ulíc M . Križovatky sú očíslované číslami od 1 do N . Nasledujúcich M riadkov vstupného súboru obsahuje popis jednotlivých ulíc v meste. Každý z týchto riadkov obsahuje dve čísla, predstavujúce čísla križovatiek, ktoré príslušná ulica spája. Prvé číslo v každom riadku je menšie ako druhé z nich. Tieto riadky sú v súbore zotriedené podľa prvého čísla. V prípade, že má viac ulíc rovnaké prvé číslo, sú zotriedené podľa druhého. Môžete predpokladať, že každé dve križovatky spája najviac 200 ulíc.

Výstup:

Výstupný súbor **OKRUZNE.OUT** obsahuje toľko riadkov, koľko má spoločnosť prevádzkovať autobusových liniek. Každý riadok obsahuje popis práve jednej autobusovej linky. Trasa autobusovej linky je popísaná ako postupnosť čísel križovatiek, ktorými linka prechádza. Prvé a posledné číslo uvedené v riadku musia byť rovnaké (trasa je okružná, teda začína a končí na tej istej križovatke). Jednotlivé čísla sú v každom riadku oddelené práve 1 medzerou. Pokiaľ sa trasa liniek nedá navrhnuť tak, aby vyhovovali podmienkam zo zadania úlohy, potom výstupný súbor má obsahovať jediný riadok s textom „Neda sa“.

Príklad 1:

Vstup:

5 12
1 2
1 2
1 2
1 2
1 2
1 3
1 3
2 3
2 5
3 4
3 5
3 5
4 5

Výstup:

1 2 3 1
2 1 2
3 4 5 3
2 5 3 1 2

Príklad 2:

Vstup:

3 4
1 2
1 2
1 3
2 3

Výstup:

Neda sa

SLOVENSKÁ KOMISIA MATEMATICKEJ OLYMPIÁDY

50. ROČNÍK MATEMATICKEJ OLYMPIÁDY

Zadania celoštátneho kola kategórie P

Vydala IUVENTA – zariadenie pre voľný čas detí, mládeže i dospelých MŠ SR
pre vnútornú potrebu Ministerstva školstva SR
Programom T_EX sadzbu pripravil Michal Forišek

Autori príkladov: Jan Kára
Daniel Král
Martin Mareš

© Slovenská komisia matematickej olympiády, 2001