Zadání soutěžních úloh

Kategorie žáci a mládež

Soutěž v programování – 27. ročník
Krajské kolo 2012/2013
18. až 20. dubna 2013

Úlohy můžete řešit v libovolném pořadí a samozřejmě je nemusíte vyřešit všechny. Za každou úlohu můžete dostat maximálně 10 bodů, z nichž je většinou 9 bodů vyhrazeno na ohodnocení funkčnosti programu, jeho shody se zadáním a efektivity a jeden bod na dokumentaci a přehlednost zdrojového kódu. Body získané za každou úlohu se ještě násobí koeficientem, který odráží složitost úlohy.

Na řešení úloh máte 4 hodiny čistého času.

Před zahájením soutěže vám pořadatel oznámí, kde najdete testovací soubory. Textové testovací soubory pro všechny operační systémy používají řádky ukončené dvojicí znaků CR a LF.

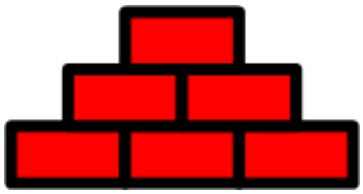
Pyramidy

Koeficient 1

Mladý Ramses se rozhodl nechat si postavit pyramidu. Jelikož se nemohl rozhodnout, jak by měla být vysoká, rozhodl se nejdříve si vytvořit plochou maketu, aby si udělal lepší představu.

Pomozte Ramsesovi a napište program, který při zadání výšky pyramidy v rozmezí 1 až 500 spočte, kolik je potřeba kvádru na její stavbu, a pyramidu vykreslí. Program by měl při vykreslení optimalně využít plochu okna.

Kvádry mají rozměry 2:1 a řady se o půl kvadrů překrývají. Například pro plošnou pyramidu výšky 3 je třeba šest kvádrů.



Odstraňování bugů

Koeficient 3

Odstraňování bugů z programu může být velmi vyčerpávající práce. Velká vědecká kapacita doktor Z. Ounek proto hledá způsob, jak by bylo možné odstraňovat bugy automaticky. Protože je ale příliš zaměstnán přemítáním o tom, jak bude slavný až se mu to podaří, zůstala implementace zkušební verze Dr. Z. Ounkova odstraňovače na vás.

Napište program, který dostane dva textové řetězce *BUG* a *PROGRAM*. Délka každého z nich je nejvýš 200 000 znaků a oba obsahují pouze číslice a velká a malá písmena anglické abecedy. Vaším úkolem je odstranit všechny výskyty řetězce *BUG* z *PROGRAM*u. Pozor na to, že odstraněním jednoho *BUG*u může vzniknout další.

Popis vstupu

Vstup načtěte ze souboru `bugy.txt` v aktuálním adresáři.

Soubor `bugy.txt` obsahuje dva řádky. První obsahuje řetězec *BUG* a druhý řetězec *PROGRAM*. Délka každého z nich je nanejvýš 200 000 znaků a oba obsahují pouze číslice a velká a malá písmena anglické abecedy. Oba řádky jsou ukončeny dvojicí znaků `CR` a `LF`.

Popis výstupu

Výstup vypište do souboru `vystup.txt` v aktuálním adresáři.

Na jediný řádek souboru `vystup.txt` vypište řetězec, který zůstane z řetězce *PROGRAM* po odstranění všech *BUG*ů. Řádek ukončete dvojicí znaků `CR` a `LF`. Pozor na to, že odstraněním jednoho *BUG*u může vzniknout další.

Příklad vstupu a výstupu

bugy.txt	vystup.txt
bug bububuggbak	bubak

Ukázková data

V adresáři `bugy` se nachází ukázková data včetně příkladu ze zadání. Pro získání plného počtu bodů by měl váš program vyřešit každá z nich během vteřiny.

Spojnicový graf

Koeficient 3

Náš oblíbený strýček Pompo by od vás potřeboval pomoci. Každou chvíli dostane mailem soubor, ve kterém je vidět, jak se mu měnil stav na bankovních účtech. Bohužel se v něm špatně orientuje a tak by mu pomohlo, když by se mohl podívat, zda mu peníze přibývají či ubývají. Napiště pro strýčka Pompa program, který převede přijatý textový soubor na spojnicový graf.

Uvažujte, že v jednom souboru mohou být informace o jednom až pěti bankovních účtech. Všechny pak vykreslete do jednoho grafu a pro jejich rozlišení použijte různé barvy. Nezapomeňte na legendu (popis grafu).

Vstupní soubor

Na vstupu je textový soubor, který obsahuje v prvním řádku hlavičku, v dalších řádcích pak jednotlivé hodnoty. Hodnoty na řádku jsou mezi sebou odděleny znakem středník. První sloupec obsahuje vždy datum, další sloupce pak jednotlivé zůstatky na účtech k danému datu. Zůstatky jsou vždy celočíselné. Hlavička obsahuje texty bez diakritiky a její první položka vždy obsahuje řetězec `Datum`. Umožněte uživateli, aby si vstupní soubor mohl sám vybrat po spuštění programu.

Vzhled grafu

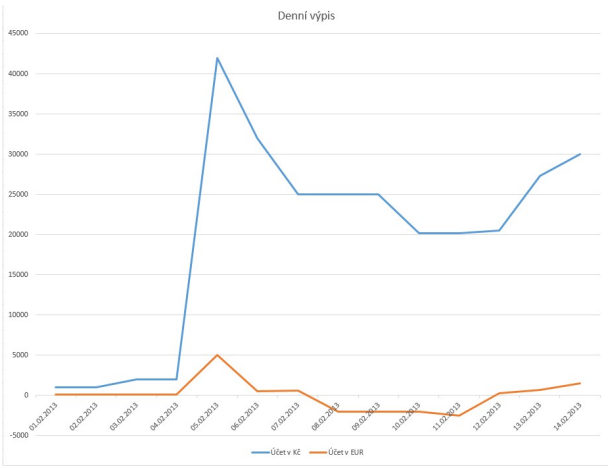
- Osa X bude zachycovat datum.
- Osa Y bude zachycovat stav na účtu (uvažujte, že stav na účtu může být dočasně i záporný).
- Vykreslený graf by měl optimálně vyplnit celou plochu obrazovky.
- Vykreslený graf vždy obsahuje legendu.

Příklad vstupu a výstupu

Vstupní soubor `denni_vypis.csv`

```
Datum;Ucet v Kc;Ucet v EUR;
1.2.2013;1000;100;
2.2.2013;1000;105;
3.2.2013;2000;105;
4.2.2013;2000;105;
5.2.2013;42000;5000;
6.2.2013;32000;500;
7.2.2013;25000;600;
8.2.2013;25000;-2000;
9.2.2013;25000;-2000;
10.2.2013;20195;-2000;
11.2.2013;20195;-2500;
12.2.2013;20542;300;
13.2.2013;27329;700;
14.2.2013;30037;1500;
```

Vykreslený graf



Vstupní soubor `mesicni_vypis.csv`

Vykreslený graf

```
Datum;Ucet v CSOB;Ucet v KB;Ucet v CS;
31.1.2013;159123;1000123;312345;
28.2.2013;234576;2001239;300986;
31.3.2013;20432;3002561;253612;
```