

Malý google

Termín odevzdání:	17.12.2017 23:59:59
Hodnocení:	0.0000
Max. hodnocení:	5.0000 (bez bonusů)
Odevzdaná řešení:	0 / 20 Volné pokusy + 10 Penalizované pokusy (-10 % penalizace za každé odevzdání)
Nápovědy:	0 / 2 Volné nápovědy + 2 Penalizované nápovědy (-10 % penalizace za každou nápovědu)

Úkolem je realizovat program, který bude fungovat jako "malý google" - bude vyhledávat zadané dotazy v textu.

Předpokládáme text zadaný na vstupu. Text obsahuje slova (tvořená písmeny malé a velké abecedy) a oddělovače slov (všechny ostatní znaky - mezery, tabulátory, odřádkování, interpunkci, číslice, ...). Text je organizovaný do řádek a je zadaný na standardním vstupu. Program tento vstup načte do svých paměťových struktur. Zadávaní textu končí po zadání prázdné řádky.

Po zadání textu jsou zadávány jednotlivé dotazy. Jeden dotaz je zadaný na jedné řádce. Dotaz je tvořen jedním nebo více slovy (znaky malé a velké abecedy), ostatní znaky slouží jako oddělovače slov. Program zpracuje dotaz a vyhledá v textu místa, kde se za sebou nachází zadaná hledaná slova. Při hledání nás zajímají pouze slova, ostatní znaky (oddělovače slov) přeskakujeme. Při porovnávání slov se nerozlišují malá a velká písmena. Hledaná slova jsou nalezena, pokud:

- v textu jsou obsažena všechna hledaná slova z dotazu,
- pořadí slov v textu odpovídá pořadí slov v dotazu,
- mezi hledanými slovy nejsou vložena další slova navíc,
- hledaná slova jsou nalezena jako celá slova.

Pokud je zadaný dotaz například:

`Lorem ipsum`

Příklady, kde je hledaný text nalezen:

`zakladni lorem ipsum verze`

`oddelovace nehraji roli lorem, ipsum`

`oddelovace nehraji roli 'lorem, ipsum'`

`oddelovace nehraji roli lorem , 123456 : 456456 ipsum`

`oddelovace nehraji roli 12lorem34ipsum56`

`nezalezi na malych a velkych pismenech LOREM ipsum`

`hledany text muze byt rozdelen do vice radek jako zde lorem
ipsum dolor sit amet`

Příklady, kde je hledaný text není nalezen:

`neshoda v nekterem ze slov lorem ipsun`

`chybi nektere ze slov lorem dolor`

`vlozene slovo navíc lorem amet ipsum`

`hledane slovo nenalezeno jako cele slovo lorem ipsumdolor`

`nesouhlasí poradi slov ipsum lorem`

Vstup programu je nejprve prohledávaný text, následuje prázdná řádka a pokračuje seznam dotazů (jeden dotaz na řádku). Zadávaní dotazů končí s koncem vstupu (EOF).

Výstupem programu je informace, zda byla zadaná slova nalezena nebo ne. Pokud jsou zadaná slova nalezena, je zobrazen seznam čísel řádek, kde lze hledaná slova najít. Pokud jsou hledaná slova rozdělena do více řádek, je v seznamu

uvedeno počáteční číslo řádky. Pokud jsou hledaná slova nalezena na řádce vícekrát, je ve výsledném seznamu tato řádka uvedena vícekrát (odpovídající počtu nalezených výskytů). Číslo řádek na výstupu tvoří neklesající posloupnost.

Pokud je vstup neplatný, program to musí detekovat, zobrazit chybové hlášení a ukončit se. Chybové hlášení zobrazujte na standardní výstup (ne na chybový výstup). Za chybu považujte:

- není správně zadáný prohledávaný text, tedy pokud na vstupu chybí prázdná řádka ukončující zadávání prohledávaného textu,
- prohledávaný text je prázdný (neobsahuje žádná slova).

Pokud bude zadáný prázdný dotaz (neobsahuje žádná slova), program vypíše, že dotaz není platný (viz ukázka), ale bude pokračovat ve zpracování dalších dotazů.

Program je testován v omezeném prostředí. Je omezena doba běhu a dostupná paměť. Časové i paměťové limity jsou nastavené tak, aby jimi prošla správná implementace postavená na naivním algoritmu. Paměťový limit je navíc navýšen tak, aby poskytoval rezervu pro uložení předpočítaných dat pro případný efektivnější algoritmus hledání. Na druhou stranu, paměťový limit zohledňuje velikost vstupů. Určitě neprojde řešení, které staticky alokuje paměť např. pro 1000000 znaků vstupního textu. Úloha nabízí bonusové testy. V prvním bonusovém testu je zadán dlouhý vstupní text, ve kterém je mnoho míst s potenciální shodou s hledanými slovy. Druhý bonusový test je obdobný, navíc je na vstupu obrovské množství dotazů nad tím samým prohledávaným textem. Pro úspěšné zvládnutí bonusových testů je potřeba použít efektivní algoritmus a předzpracovat si vstupní data.

Ukázka práce programu:

Text:

```
Lorem ipsum dolor sit amet, consectetur adipiscing elit.
Mauris dictum facilisis augue. Etiam neque. Morbi scelerisque id
luctus id velit. Praesent id justo in neque elementum ultrices.
lorem
+
ipsum
```

Hledani:

```
ipsum
Nalezeno: 1, 6
lOrEm
Nalezeno: 1, 4
id
Nalezeno: 2, 3, 3
um
Nenalezeno
amet   consectetur
Nalezeno: 1
elit Mauris
Nalezeno: 1
id      ,      luctus
Nalezeno: 2
Etiam Morbi
Nenalezeno
```

Neplatny dotaz

```
lorem ipsum
Nalezeno: 1, 4
```

Text:

```
Lorem ipsum dolor sit amet, consectetur adipiscing elit.
Mauris dictum facilisis augue. Etiam neque. Morbi scelerisque id
luctus id velit. Praesent id justo in neque elementum ultrices.
Nespravny vstup.
```

Nápověda:

- Ukázkové běhy zachycují očekávané výpisy Vašeho programu (tučné písmo) a vstupy zadané uživatelem (základní písmo). Zvýraznění tučným písmem je použito pouze zde na stránce zadání, aby byl výpis lépe čitelný. Váš program má za úkol pouze zobrazit text bez zvýrazňování (bez HTML markupu).

Vzorová data:

[Download](#)

☐ Referenční řešení