Zadání úlohy do projektu z předmětu IPP 2013/2014

Zbyněk Křivka a Dušan Kolář
E-mail: {krivka, kolar}@fit.vutbr.cz, {54 114 1313, 54 114 1238}

CST: C Stats

Zodpovědný cvičící: Patrik Šebeň (iseben@fit.vutbr.cz)

1 Detailní zadání úlohy

Vytvořte skript pro analýzu zdrojových souborů jazyka C podle standardu ISO C99, který ve stanoveném formátu vypíše statistiky komentářů, klíčových slov, operátorů a řetězců.

Tento skript bude pracovat s těmito parametry:

- `--help` viz společné zadání všech úloh
- `--input=fileordir` zadaný vstupní soubor nebo adresář se zdrojovým kódem v jazyce C. Uvažujte soubory v kódování ISO-8859-2. V případě zadání konkrétního souboru je analyzován tento soubor bez ohledu na jeho příponu. Jeli zadán adresář, tak jsou postupně analyzovány všechny soubory s příponou jazyka C (`.c`, `.h`) v tomto adresáři a všech jeho podadresářích. Pokud nebude tento parametr zadán, tak se analyzují soubory (opět pouze s příponou `.c` a `.h`) z aktuálního adresáře a všech jeho podadresářů. Je-li zadán adresář a některý z analyzovaných souborů (opět pouze s příponou `.c` a `.h`) nelze číst, ukončete skript s chybovým hlášením a návratovým kódem 21.
- `--nosubdir` prohledávání bude prováděno pouze v zadaném adresáři, ale už ne v jeho podadresářích. Parametr se nesmí kombinovat s případem zadání konkrétního souboru pomocí parametru `--input`.
- `--output=filename` zadaný textový výstupní soubor v kódování ISO-8859-2 (přesný formát viz níže)
- `-k` vypíše počet všech výskytů klíčových slov (vyskytujících se mimo poznámky a řetězce) v každém zdrojovém souboru a celkem (pro všechny analyzováné soubory)
- `-o` vypíše počet výskytů jednoduchých operátorů (nikoliv oddělovačů apod.) mimo poznámky, znakové literály a řetězce v každém zdrojovém souboru a celkem (pro všechny analyzováné soubory). Jednoduchý operátor definujme jako danou dopředu známou a fixní posloupnost nepísmenných znaků[1].
- `-i` vypíše počet výskytů identifikátorů (mimo poznámky, znakové literály a řetězce) v každém zdrojovém souboru a celkem (pro všechny analyzováné soubory) – nezahrnuje klíčová slova

[1] Jednoduchý operátor tedy není např. `sizeof`, operátor přetypování, indexace nebo volání funkce. Obdobně v základním zadání neuvažujte za operátor čárku. Není třeba uvažovat digraphy, trigraphy (viz norma jazyka), ani operátory používané v makrech (`#`, `##`).

- `-w=pattern` vyhledá přesný[2] textový řetězec `pattern` ve všech zdrojových kódech a vypíše počet nepřekrývajících se výskytů na soubor i celkem. Jelikož se nejedná o identifikátor ale řetězec, hledá se i v poznámkách, makrech, znakových literálech a řetězcích! Toto vyhledávání je case-sensitive.
- `-c` vypíše celkový počet znaků komentářů včetně uvozujících znaků komentářů (`//`, `/*` a `*/`) na soubor a celkem (pro všechny analyzované soubory). Komentáře počítejte včetně znaku konce řádku[3] uvnitř blokového komentáře a v případě řádkového komentáře také započítejte znak konce řádku[3] do počtu znaků komentáře.
- `-p` v kombinaci s předchozími (až na `--help`) způsobí, že soubory se budou vypisovat bez úplné (absolutní) cesty k souboru (tedy pouze samotná jména souborů; řazení souborů pak také neuvažuje absolutní cesty k souborům a řadí pouze podle jména souborů).

Parametry `-k`, `-o`, `-i`, `-w` a `-c` nelze mezi sebou kombinovat a pokud nebude uveden `--help`, tak je požadováno uvedení právě jednoho z těchto parametrů. Parametr `-p` je samozřejmě možné kombinovat s uvedenými parametry. Formát výpisu bude následující (vypisované záznamy souborů jsou seřazeny vzestupně, viz dále):

```
<jméno souboru příp. i s absolutní cestou>  <správné odsazení>  <číslo>
<jméno souboru příp. i s absolutní cestou>  <správné odsazení>  <číslo>
<jméno souboru příp. i s absolutní cestou>  <správné odsazení>  <číslo>
...                                         ...                 ...
CELKEM:                                     <správné odsazení>  <číslo>
```

Přitom `<správné odsazení>` je takový přesný minimální počet mezer, aby všechna čísla končila na stejném sloupci (první sloupec je zarovnán doleva a poslední sloupec doprava) a žádné nezačínalo ve sloupci začínajícím za tím sloupcem[4], kde se, byť na jiném řádku výpisu, vyskytuje text jména souboru. Výstup je ukončem odřádkováním. Seznam souborů bude lexikograficky seřazen (vzestupně podle ordinálních hodnot písmen v ASCII tabulce). Názvy analyzovaných souborů (ani cesty k nim) nebudou obsahovat žádnou diakritiku.

Skript bude prohledávat všechny soubory od místa uložení hlouběji (myšleno k podadresářům), které obsahují soubor s platným rozšířením (pro jazyk C to budou `.c` a `.h`). POZOR! Testovací adresářová struktura se zdrojovými texty jazyka C, ale případně i jinými soubory, které je třeba ignorovat, bude ke skriptu nahrávána námi automaticky. Všechny analyzované zdrojové soubory budou obsahovat validní zdrojový kód jazyka C (dle normy ISO/IEC 9899:TCC).

Makra preprocesoru jazyka C ignorujte (např. vypuštěním před samotným zpracováním zdrojového textu), což znamená, že definice maker budou vynechány podobně jako komentáře (nezapočítávají se však jako komentáře!). Volání maker ve zbylém zdrojovém kódu nijak speciálně neřešte (většinou se volání makra rozpozná jako identifikátor, což je lexikálně v pořádku). Pozor na fakt, že definice makra může pokračovat i na následujícím řádku při použití znaku \ na konci řádku s definicí makra.

Reference:

- Standard ISO/IEC 9899:TCC Committee Draft — September 7, 2007. Dostupné na `http://www.open-std.org/jtc1/sc22/wg14/www/docs/n1256.pdf` [citováno 1.2.2011]

[2] Řetězec `pattern` nereprezentuje regulární výraz, ale pouze čistý text. Obsahuje-li jednu či více mezer, tak bude ohraničen apostrofy.

[3] Započítávejte `\n` stejně jako `\r\n` obojí jako jeden znak.

[4] Kromě nekolize sloupců je třeba vložit také minimálně jednu mezeru.

2 Bonusová rozšíření

Jako bonusové rozšíření lze uvažovat přesnější (v rámci možností) statistiku nejednoduchých operátorů, které se budou započítávat nepovinným parametrem `-s` v kombinaci s `-o`. Rozšíření bude hodnoceno následujícím způsobem:

- **TER** (až 1 bod): Ternární operátor ? : v kombinaci se správnou identifikací operátoru čárka.
- **IND** (až 0,5 bodu): Správné rozpoznávání netriviálních použití operátorů indexace.
- **FNC** (až 0,5 bodu): Volání funkcí.

Další rozšíření se na rozdíl od předchozích používá v kombinaci s parametrem `-c`.

- **COM** (až 0,5 bodu): Započítávání znaků komentáře i v rámci definic makra.

3 Poznámky k hodnocení

Výstupy budou zpracovány vlastním nástrojem na porovnání výsledků s očekávanými výsledky i na přesně požadované formátování výsledků, takže každý znak i mezera při zarovnání výsledků jsou důležité.

Upozornění: V závislosti na kontextu může stejný řetězec v jazyce C odpovídat různým typům lexémů (např. `*` může znamenat operátor násobení, operátor dereference, nebo součást deklarace, kdy se o operátor nejedná).