Dokumentace úlohy CST: C Stats v Pythonu 3 do IPP 2014/2015
Jméno a přijmení : Jakub Heger
Login: xheger00

Skript pracuje se zdrojovými soubory jazyka C (ISO C99), které analyzuje a vypíše pro ně statistiky komentářů, klíčových slov, operátorů, identifikátorů nebo řetězců. Lze jej rozdělit do 4 hlavních částí: zpracování parametrů skriptu, otevírání souborů, analýza souboru a výpis do souboru nebo na standartní výstup.

1. Zpracování parametrů

Ke zpracování parametrů používám vlastní třídu `MyParser` která je vytvořená ze třídy `argparse` (součást knihovny `argparse`). Důvod je ten, že mi to dovoluje vypisovat vlastní formát chyb. Skript zkontroluje jestli nebyl spuštěn s nepovolenou kombinací parametrů, případně se ukončí s chybovým kódem 1.

2. Otevírání souborů

Pokud byl zadán parametr `--input` je jeho obsah převeden do absolutní cesty, jestli ne tak se použije cesta do adresáře ze kterého je skript spouštěn. Tato cesta je poslána funkci `getFileList` která získá cesty k validním souborům (s příponami .c nebo .h). V případě že nebyl zadán parametr `--nosubdir` volá se funkce `getFileList` rekurzivně dokud neprojede všechny podadresáře, jinak se prochází pouze zadaný adresář. Cesty k validním souborů jsou uloženy do globální proměnné `listOfFiles`. Pokud je cílem jeden konkrétní soubor, uloží se cesta k němu bez ohledu na jeho příponu. Následně jsou všechny soubory v proměnné `listOfFiles` otevřeny a jejich obsahy uloženy do proměnné `contentOfFiles`. Jestliže došlo při otevírání souborů k chybě tak se skript ukončí s chybou (pokud byl cíl adresář ukončí se s kódem 21, jinak 2).

3.Analýza souboru

Před analýzou se nejprve projdou obsahy souborů a všechny výskyty dvojice znaků `\` a `'\n'` jsou nahrazeny prázdným řetězcem.

-a) Parametr -w=PATTERN

Spočítá počet výskytů PATTERNu v analyzovaných souborech pomocí knihovní funkce `string.count` a výsledky pro každý soubor postupně ukládá do globální proměnné `resultsList` (stejné pro jakýkoli parametr).

-b) Parametr -c

Pro každý soubor zavolá funkci `stripComments` která odstraňuje komentáře a zároveň počítá kolik znaků obsahují. Řádkové komentáře nalezne pomocí řídící sekvence `'//'` a spočítá/odstraní zbytek řádku včetně. U blokových komentářů spočítá/odstraní vše mezi sekvencemi `'/*'` a `'*/'` včetně. Implementováno včetně rozšíření COM, takže skript počítá i komentáře uvnitř definic maker preprocesoru.

-c) Parametr -k

Z obsahů souborů se nejprve odstraní řetězce (vše mezi znaky `""`, funkce `stripStrings`), komentáře a makra prepocesoru (od znaku `#` do konce řádku, funkce `stripMacros`). Poté se pomocí regulární výrazů vyhledají a spočítají výskyty všech klíčových slov uložených v seznamu `keywords`.

-d) Parametr -i

Z obsahů souborů se nejprve odstraní řetězce, znakové literály (vše mezi znaky `''`, funkce `stripLits`) komentáře, makra a klíčová slova (stejným způsobem jako se počítal jejich výskyt, funkce `stripKeywords`). Vše co v souboru zbyde a odpovídá pravidlům pro pojmenování identifikátorů je spočítáno a uloženo.

-e) Parametr -o

Z obsahů souborů se nejprve odstraní řetězce, znakové literály, deklarace pointerů a číselné konstanty (pomocí regulárních výrazů, funkce `stripDeclarations`), komentáře a makra preprocesoru. Poté se naleznou, spočítají a odstraní operátory ze seznamu `operatorsRegexps`. Tyto operátory jsou seřazeny od nejdelších po nekratší aby nedocházelo ke kolizím a tedy ke špatnémů spočítání (například odstranění `*` z operátor `*=` by způsobilo započítaní jednoho operátoru dvakrát.). Regulární výraz pro operátor dereference je napsán tak aby i při vícenásobném použití byl započítán pouze jednou. Operátor pro přístup ke členu struktury naopak musí být jen jednou za sebou, aby

nezapočítával elipsis.

5.Výpis

Pokud byl nastaven parametr `-p` odstraní se z `listOfFiles` absolutní cesty a zůstanou pouze jména souborů. Poté se spojí obsahy seznamů `listOfFiles` a `resultsList`. Ve výsledném seznamu se nalezne nejdelší řádek a podle něj se přidá správné odsazení do ostatních řádku. Pak se seznam seřadí a nakonec se přidá řádek CELKEM: a součet všech výsledků. Jestli není nastaven parametr `--ouput` provede se tisk na standartní výstup, jinak se otevře soubor (pokud se to nepodaří skript se ukončí s kódem 3). Poté se soubor uzavře a skript se standartně ukončí.

6. Známé problémy

Skript nerozpozná složitější deklarace pointerů a hrozí tedy zaměnění `*` v deklaraci za operátor. Taktéž dokáže rozpoznat pouze několik základních typů číselných konstant a může tedy započítat desetinou tečku či `+/-` u exponentu jako operátor.

Nahrazení `\\n` za prázdný řetězec se provede v každém případě a tedy i když není zpětné lomítko platné (součást komentáře nebo řetězce).