

Zadání úlohy

Úkolem této úlohy bylo vytvořit skript pro analýzu zdrojových souborů jazyka C podle standardu ISO C99. Výstupem skriptu je výpis statistiky komentářů, klíčových slov, řetězců nebo operátorů dle zadaného vstupního parametru. Tato dokumentace není celistvým popisem řešené úlohy, pouze popisuje způsob implementace. Pro bližší seznámení s úlohou bych případného čtenáře na oficiální zadání úlohy.

Způsob řešení

Kontrola parametrů

Na začátku celého skriptu se provádí kontrola parametrů zadaných při spuštění skriptu z příkazové řádky. Kontrola parametrů je postavena na funkci `getopt` ve spojení s vnitřní proměnnou `argc`. Při analýze parametrů se také nastavují pomocné proměnné pro určující vstupní soubor(soubory), výstupní soubor.

Načtení souborů

Podle zadaných parametrů je potřeba řešit tři různé situace. Načtení konkrétního souboru je, načtení všech souborů s koncovkou `.c` nebo `.h` v zadaném adresáři nebo všechny soubory s koncovkou `.c` nebo `.h` v daném adresáři a ve všech jeho podadresářích. Soubory jsou načteny do asociativního pole. klíčem obsahu souboru je název daného souboru.

Analýza souborů

Vzhledem k tomu, že zadání dovoluje neřešit různé speciální a složitější konstrukce, je analýza postaveno na funkcích pracujících s regulárními výrazy. Po načtení souborů následuje analýza souborů podle zadaného parametru.

- `-w` Spočítá počet zadaného řetězce. Hledaný řetězec se může nacházet v celém zdrojovém textu (tj. i v komentářích a makrech. Řešeno pomocí funkce `preg_match_all()`.
- `-o` Spočítá počet operátorů v řetězce. Řešeno postupným odstraňováním 3-znakových, následně 2-znakových a nakonec 1-znakových operátorů a průběžným počítáním.
- `-k` Spočítá počet klíčových slov. Řešeno odstraněním výčtu klíčových slov a jejich spočtení pomocí funkce `preg_match_all()`.
- `-i` Spočítá počet identifikátorů. Řešeno odstraněním klíčových slov a následně postupně iterativní odstraňováním a sčítáním řetězců odpovídajících identifikátorům.
- `-c` Spočítá počet znaků komentářů. Řešeno rozdělením na makra, literály a řetězce a následným jejich spočítáním.

pozn. parametrum `o`, `k`, `i` předchází odstranění komentářů, maker a literálů. parametru `c` pouze maker a literálů.

Výpis výsledků

Před finálním výpisem výsledků je zapotřebí seřadit lexikograficky názvy souborů, toho je docíleno funkcí `ksort()`. Po seřazení souborů jsou získané výsledky vypsány buď do zadaného souboru nebo na standardní výstup.

Závěr

Vzhledem k tomu, že se jedná o první seznámení se skriptovacím programovacím jazykem PHP, tak mohou být použity nestandardní konstrukce. Značnou část řešení tvořilo čtení dokumentace a hledání vhodných prostředků k řešení konkrétního problému. Případné chybové stavy jsou ošetřeny, a skript vrací číslo chyby ze zadání v opačném případě končí s návratovým kódem 0. Jazyk C je značně rozsáhlý a dovoluje některé nestandardní konstrukce. Vzhledem rozsáhlosti projektu a jeho bodovému ohodnocení bylo rozhodnuto neimplementovat celý lexikální analyzátor, ale spokojit se s řešením, které nepodchycuje všechny možné stavy jazyka. Kostrou celého řešení tvoří převážně funkcemi, které pracují s regulárními výrazy.