

Implementačná dokumentácia k 1. úlohe z predmetu IPP

Autor: Simona Jánošíková (xjanos19)

Prvým krokom v implementácii skriptu `parse.php` bola implementácia funkcie `ErrorPrint (int $error_code)`, ktorá je volaná v prípade zistenej chyby za behu skriptu a jej úlohou je vypísať chybovú hlášku podľa zistenej chyby a ukončiť program s korektnou návratovou hodnotou.

Následne som spracovala a overila korektnosť vstupných argumentov programu. Skript podporuje iba jeden argument, t. j. argument `--help`. Po zadaní argumentu `--help` sa vypíše náponeda s inštrukciami ako spustiť a ovládať skript. Pre overenie korektnosti vstupných argumentov a prípadný výpis náponedy som naimplementovala funkciu `LoadArgs (int $argc, array $argv)`.

V hlavnom tele programu sa pomocou while cyklu spracováva vstupný jazyk `IPPcode23`. `IPPcode23` zahrňuje hlavne trojadresné inštrukcie, ktoré sa nachádzajú vždy na novom riadku. Hlavný while cyklus načítava vstup riadok za riadkom a z každého riadku prípadne odstráni komentáre a rozdelí slová v riadku do poľa `parsed_line`. Na overenie výskytu hlavičky som použila premennú `header` typu boolean, ktorá informovala, či už bola prečítaná hlavička alebo nie a pomocou nej som aj identifikovala výskyt hlavičky vo vstupnom jazyku. Po úspešnom spracovaní hlavičky som začala spracovávať vstup po riadkoch. V `parsed_line` sa na prvej pozícii nachádza kód operácie, ktorý sa vo funkcií `switch` zaradí do svojej skupinky a následne sa spracuje. `Switch` vo while cykle reprezentuje lexikálnu analýzu vstupného programu. Inštrukcie sú v `switch-i` rozdelené do skupiniek podľa počtu a typu operandov alebo podľa toho, čo má jednotlivá operácia vykonať. Operácia sa vo funkcií `switch` zaradí do svojej skupinky a pomocou funkcie `arg_number (array $args, int $op_args)` sa overí korektný počet argumentov danej inštrukcie. Potom sa pomocou funkcie `printArg ($xml, int $arg_numb, string $type, string $arg)` overí korektnosť argumentov príkazu, čo značí syntaktickú analýzu a v prípade jej úspešnosti sa vypíše XML výstup danej inštrukcie a jej argumentov. Pre spracovanie regexov som použila funkcie ako `preg_match`, `preg_replace`, `preg_split`, ktoré zahrňuje knižnica `PCRE`.

XML výstup som tvorila pomocou knižnice `XMLWriter`, ktorá po úspešnej kontrole hlavičky vygeneruje začiatok XML výstupu a následne sa jednotlivé operácie a ich operandy generovali vo funkcií `printArg ($xml, int $arg_numb, string $type, string $arg)` po úspešnom vykonaní syntaktickej analýzy.