

Implementační dokumentace k 1. úloze do IPP 2021/2022

Jméno a příjmení: Pavel Tetauer

Login: xtetau00

1. Úvod

Úkolem bylo implementovat skript `parse.php`, který provádí lexikální a syntaktickou analýzu zdrojového kódu v IPP-code22 a následně vypíše jeho XML reprezentaci.

2. Řešení

Výsledný skript `parse.php` na začátku zkontroluje, zda se nevyskytuje jiný argument, než jediný v tomto skriptu povolený `--help` a zda standardní vstup není prázdný. Poté vypíše na standardní výstup XML hlavičku a začne v cyklu po řádcích procházet zdrojový kód zasláný na standardní vstup.

Před samotnou syntaktickou a sémantickou kontrolou jsou nejprve z řádku odstraněny komentáře a přebytečné bílé znaky. Po této úpravě se zkontroluje, zda se na prvním řádku nachází `.IPPcode22`.

Kontrola zbývajících řádků probíhá tak, že se všechny části instrukce rozdělí pomocí funkce `explode()` a s takto rozdělenou instrukcí se následně pracuje.

Celá následná kontrola a výpis jsou implementovány pomocí příkazu `switch`. Nejprve se zkontroluje, co se nacházelo na začátku řádku. To musí být operační kód. Pokud tento operační kód program nenalezne v žádné větvi `case`, je vypsána chyba a program dále nepokračuje.

Následné operandy instrukce jsou kontrolovány tak, že se nejprve zkontroluje, zda sedí jejich počet. Pokud nesedí, dojde k chybě. Pokud ano, vypíše se XML pro začátek instrukce.

Na základě toho, jaké operandy jsou očekávány, následuje jejich zpracování v těchto funkcích:

`check_and_print_var()` – zkontroluje a v případě správnosti zapíše proměnnou

`check_and_print_label()` – zkontroluje a v případě správnosti zapíše návěští

`check_and_print_symb()` – zkontroluje, zda se jedná o správně zapsanou konstantu či proměnnou. V případě proměnné použije již zmiňovanou funkci pro práci s proměnnou, v případě konstanty použije funkci `check_and_print_const()`

Všechny výše zmiňované funkce využívají ještě odpovídající pomocné funkce `is_var()`, `is_const()`, `is_label()` a `is_symb()` pro kontrolu, zda se jedná o správný typ operandu. Na tuto kontrolu je využita funkce `preg_match()` a regulérní výrazy.

3. Chování v případě chyb

Všechny chyby jsou vypisovány pomocí funkce `print_error()`, která na základě obdrženého chybového kódu vypíše odpovídající hlášku na standardní chybový výstup. Součástí chybové hlášky je i číslo řádku, na kterém se tato chyba ve zdrojovém kódu nachází.

V případě chyby, která spočívá v argumentech, nebo chybějícím souboru na standardním vstupu, nedojde k žádnému vypsání XML reprezentace. V případě odhalení chyby ve zdrojovém kódu dojde k jejímu neúplnému vypsání.