

Implementační dokumentace k 1. úloze do IPP 2018/2019

Jméno a příjmení: Jakub Sadílek

Login: xsadil07

Úvod

Úkolem bylo vytvoření skriptu v jazyce PHP, který má provést lexikální a syntaktickou analýzu zdrojového kódu IPPcode19 a vygenerovat tento kód ve formátu XML. V případě jakékoliv chyby je skript ukončen chybovou hláškou na standardní chybový výstup a vrací odpovídající návratovou hodnotu podle zadání.

Postup a implementace

Jako první bylo potřeba zkontrolovat vstupní parametry skriptu. Jelikož skript obsahuje rozšíření STATP, může očekávat spoustu parametrů a je potřeba zajistit nejen jejich správné použití (například kolizi s parametrem pro výpis nápovědy), ale i uchovat pořadí, protože podle toho se pak budou vypisovat jednotlivé statistiky. Proto při implementaci byla použita funkce `getopt`, která práci velmi zjednodušila, i když její použití není na první pohled zcela intuitivní. Pak bylo potřeba zkontrolovat správnost shebangu IPPcode19, který musí být vždy na prvním řádku. Dále je celý proces implementován jako konečný automat, kde se neustále opakují ty stejné operace – načíst řádek, zbavit ho přebytečných mezer a komentářů, zkontrolovat operační kód a parametry instrukce včetně syntaxe daného parametru, a nakonec přidat instrukci do formátu XML. Pro syntaktickou kontrolu hodnot a názvů byly použity regulární výrazy. K zachování určité abstrakce v kódu byly implementovány dvě třídy. Jedna pro instrukce, která obsahuje informaci o operačním kódu instrukce a dva seznamy pro parametry. První seznam pro typ a druhý pro hodnotu, to znamená, že seznamy se plní současně. Dále obsahuje metody na kontrolu syntaxe. Druhá třída je pro generování XML. Třída se oproti třídě pro instrukce vytvoří pouze jednou na začátku běhu skriptu. V konstruktoru se nastaví kromě verze 1.0 a kódování UTF-8 a dalšího, také formátování výstupu přes `xml->formatOutput = true`, aby výsledné XML na výstupu bylo také vizuálně čitelné. Pro práci s XML byla využita třída `domXML`, protože mi přišla přehlednější i intuitivnější než ostatní. Důležitou metodou naší XML třídy je `add_ins`, která přidá instrukci do XML ve formátu, jak popisuje zadání. Výhodou je, že třída instrukce má všechny potřebná data už v sobě přímo uložená, tudíž nemusíme instrukci znova „rozkládat“, ale pouze tyto informace naparsujeme do XML. Předtím ale ještě je potřeba ošetřit kolizní znaky s XML jako jsou například ampersandy, protože IPPcode19 je podporuje, ale XML s nimi má problém. Poslední metodou je výpis, která se provádí na konci skriptu, pokud vše proběhlo bez problému. Při výpisu je použit parametr `LIBXML_NOEMPTYTAG`, který i prázdné hodnoty (jako to může být například u stringu) uzavře mezi dva tagy místo jednoho.

Rozšíření

Implementace nebyla složitá, jak se na první pohled zdálo. Po ověření vstupních parametrů funkce vrátí pravdivostní hodnotu, zda jsou statistiky požadovány či naopak nejsou. Tato hodnota je na konci testována a v závislosti na ní, se vypisuje. Jednotlivé informace jsou ukládány v globálních proměnných včetně souboru pro výpis. Výpis je pak seřazen ve stejném pořadí jak byly zadány argumenty.