

## Implementační dokumentace k 1. úloze do IPP 2019/2020

Jméno a příjmení: Jan Lorenc

Login: xloren15

### parse.php

Úkolem tohoto skriptu je načíst ze standardního vstupu zdrojový program v jazyce IPPcode20, provést nad ním lexikální a syntaktickou analýzu, převést ho do XML formátu a tento poslat na standardní výstup.

Pro implementaci programu jsem se rozhodl řídit se objektově orientovaným přístupem a celkem mi k tomu posloužilo pět následujících tříd.

Třída `Parser` slouží jako hlavní řídicí prvek skriptu. Obsahuje metodu `Run()`, která je ekvivalentem spouštěcích funkcí typu `main()` v jiných programovacích jazycích. V obecnosti pracuje tak, že zkontroluje vstupní parametry, poté ve smyčce načítá a kontroluje řádky, klasifikuje instrukce a následně je převádí do XML, které na konci vytiskne na standardní výstup. K většině práce využívá ostatní třídy, jejichž instance má jako své atributy, tedy zejména ke kontrole argumentů programu, práci s instrukcemi a převodem na XML. Sama obsahuje například pomocnou metodu `CheckLineContent($line)`, která vstupní řádek očesá o komentáře a přebytečné bílé znaky, zbyde tedy jen instrukce s parametry. Tu rozdělí na jednotlivé prvky a předá metodě `ClassifyInstruction($parts)`, která na základě kousků instrukce rozpozná, o kterou se jedná, a následně vytvoří instanci třídy, která tuto instrukci reprezentuje.

`Args` je třída, která načítá a rozpozná argumenty programu. Kontroluje především jejich správný formát a kombinace. Hlásí chybu, pokud parametr `-help` není samotný nebo se vyskytuje vícekrát, podobně se nesmí vícekrát zadat parametr `-stats`. Ostatní statistické parametry se mohou opakovat a do zadaného souboru se vytisknou v uvedeném pořadí.

Pokud projde validace parametrů i vstupního řádku třídami `Args` a `Parser`, reprezentují danou instrukci třídou `Instruction`. V konstruktoru dostane zadané jméno a pořadí a jinak obsahuje pouze jednu významnější metodu, kterou je `AddArgument($type, $value)`. Ta dostane v parametru skutečnou hodnotu z IPPcode20 a porovná ji s typem, který očekává. Mimo typu argumentu kontroluje i hodnotu, tedy např. `int@ahoj` validací neprojde.

Poslední třídou implementující povinné zadání je `XmlStream`. Tato třída se stará o vytváření XML formátu a staví na PHP třídě `DOMDocument`. V konstruktoru vytvoří povinnou hlavičku a kořenový element `<program>`, dále v jednoduché metodě `In($instruction)` přidá do DOMu element reprezentující instrukci a metodou `Out()` tiskne na standardní výstup.

Pátou zmíněnou třídou je `Stats`, která implementuje rozšíření STATP. Není vůbec složitá, prakticky obsahuje jen privátní atributy a k nim tzv. gettery a settery. Settery se od běžných liší v tom, že pouze navyšují svůj atribut o jedničku, nelze jim tedy přiřadit libovolnou hodnotu. Další specialitou je setter `LabelsUp($label)`, který navýší čítač pouze v případě, pokud se dané návěští v programu ještě nevyskytuje, kontroluje tedy unikátnost.