

Implementační dokumentace k projektu do IPP 2017/2018
Jméno a příjmení: Jan Kočí
Login: xkocij01

parse.php

První částí úlohy je analyzátor lexikální a syntaktické správnosti kódu jazyka IPPcode18, který na standardní výstup vypíše XML reprezentaci programu. Analyzátor je implementován ve skriptu `parse.php`, používá však třídy implementované v souborech `instruction.php`, `ArgParser.php` a také regulární výrazy ze souboru `my_regex.php`.

Skript prochází kód ze standardního vstupu po řádcích dokud nenarazí na znak konce souboru (EOF) a v každé iteraci přiřadí danou instrukci do kategorie podle počtu a typu argumentů, které očekává. V dané kategorii pak dojde ke kontrolám správného počtu argumentů a jejich správného formátu pomocí regulárních výrazů. Například regulární výraz pro návěští, symbolickou konstantu, proměnnou atd...

Po každé kontrole je následně vytvořen objekt `Instruction`, ten je inicializován jeho daty (argumenty) a je přidán do pole všech instrukcí programu.

Poté co se takto projde celý program, projdeme pole objektů typu `Instruction` a pro každý vygenerujeme jeho XML reprezentaci.

Skript také podporuje rozšíření STATP, kdy do předem zadaného souboru vypisuje statistiky analýzy.

instruction.php

Implementuje třídu `Instruction`, která reprezentuje jednu instrukci jazyka IPPcode18.

Obsahuje daný typ instrukce a pole jejích argumentů, což jsou objekty typu `Argument`, obsahující typ a hodnotu.

ArgParser.php

Třída implementující parser argumentů příkazové řádky v jazyce php. Třída je inspirována funkčností modulu `argparse` jazyka python.

interpret.py

Druhou částí úlohy je interpret XML reprezentace kódu jazyka IPPcode18. Program čte XML ze souboru zadaného parametrem `-source` a postupně provádí interpretaci všech instrukcí.

Program je implementován v souborech `interpret.py`, `program.py`, `xmlParser.py`, `MyException.py`. Ze všeho nejdříve program projde vstupní XML soubor, zkontroluje jeho správnost a jednotlivé instrukce uloží za sebe do seznamu objektů typu `Instruction`. Toto je implementováno v souboru `xmlParser.py`, jako objekt `XmlParser`, jehož metoda `parse` vrací daný seznam všech takto převedených instrukcí.

Dalším krokem je již samotná interpretace instrukcí. Vytvoříme objekt `Program`, který inicializujeme seznamem instrukcí programu. Následně zavoláme jeho metodu `Program.interpret`, ve které se celá interpretace odehrává.

Třída `Program` implementuje metodu pro každou instrukci jazyka IPPcode18. Při interpretaci prochází cyklem přes seznam instrukcí a pro každou zavolá odpovídající metodu, které předá její argumenty.

Paměťový model

Hodnoty proměnných během interpretace jsou ukládány do rámců, které jsou reprezentovány pomocí slovníků v objektu `Program`. Jejich obsah tak vlastně vytváří stav programu. Proměnné do slovníků ukládáme tak, že klíčem je její název a hodnotou je objekt typu `Variable`, který obsahuje její datový typ a hodnotu.

test.php

Program pro automatické testování funkčnosti analyzátoru a interpretu. Po vyhodnocení testů vygeneruje HTML soubor s jejich výsledky.