Implementační dokumentace k 1. úloze do IPP 2023/2024

Jméno a příjmení: Adam Ližičiar

Login: xlizic00

Riešenie projektu obsahuje 6 tried, ktoré sú riadené zo skriptu `parse.py`. Je zvolený princíp postupného prechodu zo vstupného IPPcode24 kódu do XML kódu pomocou čítania „riadok po riadku“ zo STDIN, t.j. priebežného ukladania inštrukcií do triedy `Instruction` (v nej prebehne kontrola syntaktickej a lexikálnej správnosti inštrukcie) a následne uloženia argumentov inštrukcie do triedy `InstrArgument` (taktiež v nej prebehne kontrola syntaktickej a lexikálnej správnosti argumentu). Po úspešnom skončení rozdelenia konkrétnej inštrukcie do triedy je v hlavnej vetve programu odoslaná trieda `Instruction` do triedy `XMLGenerator`, ktorá vloží do knižnice `xml.dom.minidom` nové elementy a podelementy. Po prejdení všetkých riadkov zo STDIN je na STDOUT vypísaný XML výstup.

K tejto architektúre som sa dostal po podrobnej analýze zadania. Prvým návrhom bol program, ktorý pri prechode bez použitia tried `Instruction` a `InstrArgument` vkladá informácie priamo do triedy `XMLGenerator`. Tento návrh by bol funkčný, ale používal by iba najjednoduchšie typy tried (`Arguments`, `Error`, `Stats`). Druhý návrh bol už s triedami `Instruction` a `InstrArgument`, ale najskôr by uložil všetky inštrukcie do zoznamu (array) a ten by sa odoslal až na konci programu do generátora. Aj tento návrh by bol funkčný, avšak by trval pri dlhších kódoch dlhší čas a mohol by zabrať veľké miesto v pamäti. Preto som vytvoril návrh, ktorý obsahuje aj pokročilejšie OOP a aj priebežné ukladanie do generátora, aby program nezabral miesto v pamäti a bol čo najrýchlejší.

V riešení sú zahrnuté obidve rozšírenia. NVP je zahrnuté postupnou tvorbou triedy `Instruction` a triedy `InstrArguments` pre každú jednu inštrukciu resp. argumenty inštrukcie. Tie sú následné využité v triede `XMLGenerator` pri generovaní výsledného XML kódu. Rozšírenie STATP funguje ako samostatná trieda, ktorá postupne počas chodu programu inkrementuje jednotlivé premenné a pri viac zložitých štatistikách (napríklad `--labels`) ukladá názvy postupne do reťazca. Pri ukončení hlavného cyklu sa ešte pred vygenerovaním XML kódu do STDOUT štatistiky zosumarizujú: vypočíta sa počet `badjumps` a `fwjumps` a následné sú zapísané do používateľom určeného výstupného súboru alebo výstupných súborov. V prípade niektorých druhov štatistík som si neni istý, či som zahrnul všetky (alebo či som nezahrnul) viac operačných kódov, ako bolo potrebné. Presný popis, v ktorom druhu štatistík skoku sú zahrnuté aké operačné kódy je v súbore `Instruction.py` vo funkcii `def opCodeStatistics(self, lineArray)`.