Implementační dokumentace k 2. úloze do IPP 2022/2023
Jméno a příjmení: Dominik Hofman
Login: xhofma11

1 Obecná funkcionalita a části interpretu

Celý interpret je navržen Objektově, tedy není v něm možné nalézt proměnnou či metodu, která by nebyla součástí třídy. Jako hlavní tělo programu se chová třída `Prog`, která má za úkol nejdříve zpracovat argumenty na příkazové řádce, poté si v sobě vytvořit instance tříd pro načítání a zpracování vstupního XML souboru (seřadí elementy `instruction` podle atributu `order` a následně seřadí i jejich podřadné elementy `arg` dle čísla daného argumentu, pokud je zjištěna duplicita nebo nevalidní hodnota atributu `order`, je zobrazena chybová hláška a program je ukončen), inicializuje rámce na své počáteční hodnoty. Je využito dvojího průchodu XML souboru, první průchod uloží všechna, řeší redefinice návěští a druhý průchod je využit k postupnému načítání a zpracovávání instrukcí.

2 Implementační detaily

Třída `Prog` v sobě uchovává instanci programového čítače, díky kterému se v cyklu tvoří nové instance jednotlivých instrukcí - třída `InstructionParser`. Každá instrukce je vložena do „Instruction listu“, který v sobě uchovává informace o názvu každé provedené instrukce a její pozici ve vstupním souboru - tohoto je využitu při volání instrukce `BREAK`, která vypíše obsahy všech rámců, počet vykonaných instrukcí a v neposlední řadě také všechny volané instrukce. Následně je volána metoda `execute`, která rozhoduje o jakou instrukci se jedná a jak s ní má být naloženo, zkontrolují se její argumenty a poté je instrukce vykonána nebo je zahlášena příslušná chybová hláška a program je ukončen.

Rámce jsou tvořeny jako list[1] trojice hodnot - `(název proměnné, hodnota proměnné, typ proměnné)`, což umožňuje kontrolu datových typů, inicializace proměnných apod. Přístup k rámcům typu `LF` je umožněn díky listu rámců `LF`, přístup k poslednímu takovému vytvořenému rámci je umožněn za pomoci funkce `pop`, která je v Pythonu k dispozici a která vrátí poslední prvek listu, tedy v tomto případě poslední list `LF`. Třída `Frame` v sobě obsahuje veškerou práci s rámci, tedy přidávání proměnných, zjišťování datových typů, kontrolu definice, inicializace proměnných, modifikaci hodnot proměnných apod.

Skoky jsou řešeny pomocí změny programového čítače, který je po detekci návěští změněn na pozici daného návěští, v případě instrukcí `CALL` a `RETURN` je uložena původní hodnota programového čítače pro možnost vrátit se zpět v programu.

[1] list - Datová struktura, která umožňuje ukládat a organizovat kolekci hodnot v jednom objektu

3 UML Diagram

UML diagram byl vygenerován přímo ze zdrojového kódu pomocí nastrojů `pyreverse` a `graphviz` a následně upraven v programu `drawio`. Obsahuje v sobě veškeré třídy, jejich metody a proměnné.