

Implementační dokumentace k 2. úloze do IPP 2022/2023

Jméno a příjmení: Petr Teichgráb

Login: xteich01

1 Návrh

Interpret je objektivě navržen a skládá se z celkových 17 tříd. Větší počet tříd je zapříčiněn především, že jsem se rozhodl pro každou skupinu instrukcí implementovat vlastní třídu, tak aby byly instrukce logicky uskupené. Všechny třídy představující konkrétní instrukci dědí ze své rodičovské třídy `Instruction`. Podrobněji popsany návrh viz 1 (diagram neobsahuje všechny třídy, které dědí ze třídy `Instruction` z důvodu nedostatku místa).

2 Implementace

Veškeré řízení programu se odehrává ve třídě `Interpret`. K parsování argumentů jsem použil modul `argparse`¹. Pro zpracování XML kódu byl použit modul `xml.etree.ElementTree`². Po získání XML stromu převádím tenhle strom na list instrukcí v metodě `ConvertXmlTreeToInstructionList` a přitom kontroloji správnost XML formátu. Pro vytvoření instance správné instrukční třídy je použit návrhový vzor `factory`. Interpret je spouštěn metodou `Start`, která nejprve projde celý list instrukcí a zpracovává pouze instrukci `LABEL`, aby bylo umožněno v programu skákat dopředu. Následně je list instrukcí projit znovu a již se zpracovávají všechny instrukce vyjma `LABEL`. Interpret nad daným objektem v listu volá metodu `Execute`, která je implementována v rodičovské třídě a podle potřeby overrideována. Metoda `Execute` na základě jména instrukce (opcode), za pomoci funkce `eval`, volá metodu stejného jména, která simuluje popsané chování instrukce. Metody neobsahují žádné argumenty (kromě `self`), jelikož jsou všechny potřebné atributy a metody získávány ze třídy `InstructionArgument`. V jednotlivých metodách již probíhají typové kontroly a další logika.

Metody pro typové kontroly jsou implementovány ve třídě `TypeController` a jsou realizovány za pomoci porovnávání vstupního atributu `type` s příslušnými hodnotami a regulárních výrazů.

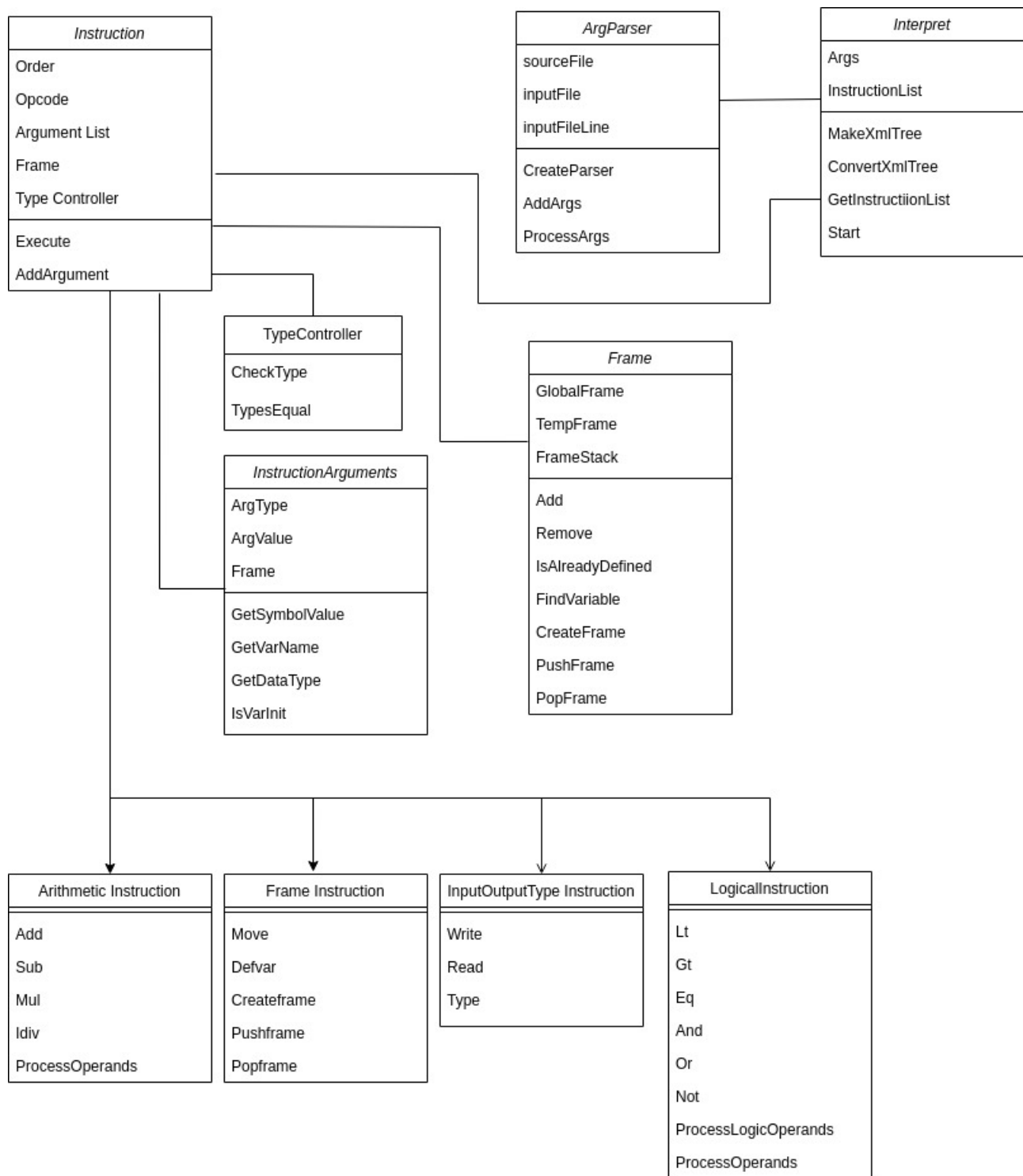
Pro práci s rámci je navržena třída `Frame`, která obsahuje metody pro práci s nimi a také atributy reprezentující všechny rámce. Rámce jsem simuloval za pomoci jednoho listu pro každý typ rámce. Do listů jsou vkládány/odebírány proměnné, které jsou definovány třídou `var`. Objekt, který je instancí třídy `Frame` je posílán jako inicializační argument do všech instrukčních tříd. Ve třídě `Frame` jsou totiž metody pro hledání proměnných a další, které jsou nezbytné ve většině instrukcí.

3 Testování

Při vývoji byl skript testován mnou vytvořenými testy a testy vytvořeny jinými studenty.

¹<https://docs.python.org/3/library/argparse.html>

²<https://docs.python.org/3/library/xml.etree.elementtree.html>



Obrázek 1: Diagram tříd interpretu