

# Interpret

## 1 Struktura programu

Hlavní funkce programu, která slouží jako jeho vstupní bod, se skládá pouze z 5 řádků/kroků. Jako první se instancuje objekt třídy `ParamHandler`. Poté se instancuje objekt třídy `XmlParser`, které jako parameter konstruktoru dáme soubor s výstupem z parseru (source parameter skriptu). A pak zavoláme metodu `parse()` objektu třídy `XmlParser`. Nasledně se instancuje třída `ProgramExecutor`, jenž jako parametry konstruktoru dostane slovník instrukcí a soubor s uživatelským vstupem programu (input parametr skriptu). Jako finalní krok se zavolá funkce `exec()` objektu třídy `ProgramExecutor`.

### 1.1 Třída `ParamHandler`

Tato třída při své instanciaci zpracuje zadané parametry skriptu pomocí python modulu `getopt`. Informace o parametrech jsou uloženy v této třídě a můžeme je odtud získat pomocí funkcí `get_source()` a `get_input()`.

### 1.2 Třída `XmlParser`

Třída `XmlParser` má na starosti rozparsování vstupního XML souboru. Práce s XML je implementována s pomocí modulu `xml.etree.ElementTree`. Při zpracovávání souboru se ověří správnost formátu XML a také struktura jeho obsahu. Pro ověření počtu a typu argumentů u jednotlivých instrukcí se hojně využívá slovníku s informacemi o instrukcích.

#### 1.2.1 Slovník s informacemi o instrukcích

Tento slovník s rozsahem v rámci celého modulu v sobě uchovává informace o všech podporovaných instrukcích. Klíči slovníku jsou jména (opcodes) daných instrukcí a hodnotami pro klíče jsou pak pole obsahující název třídy vykonávající danou instrukci, počet argumentů a pole s typy argumentů (jména tříd korespondující typům argumentů). Jeden záznam pak může vypadat takto: `'MOVE': [Move, 2, [ArgVariable, ArgSymbol]]`

### 1.3 Třída `ProgramExecutor`

Tato třída v sobě uchovává všechny potřebné věci pro běh programu (prostředí) jako například datové rámce, zásobníky a hlavně slovník se seřazenými instrukcemi. Při její instanciaci přiřadí počáteční stav všem svým proměnným a získá z instrukcí všechna návěští programu. Nejdůležitější funkcí této třídy je funkce `exec()`, která začne procházet slovník instrukcí a volat jejich funkce pro provedení. Dále obsahuje pomocné metody, které se vykonávají v jednotlivých instrukcích pro práci s datovými rámci.

## 2 Návrhový vzor: Prototyp

Návrhový vzor který je v tomto programu použit je prototyp. Tento vzor je použit při implementaci jednotlivých instrukcí. V programu máme třídu `Instruction` z které dědí všechny další třídy pro instrukce (každá instrukce má svoji korespondující třídu). V této třídě máme proměnné (`opcode`, `args[]`, metody pro jejich získání, metodu `execute()` a konstruktory, který je stejný pro všechny další třídy instrukcí. Metodu `execute()`, která slouží k provedení dané instrukce přepisujeme ve všech třídách co dědí z této třídy. Toto nám umožňuje ve třídě `ProgramExecutor` volat nad slovníkem instrukcí pouze jednu a tu samou metodu která vždy provede něco jiného záleží na dané instrukci. Také využíváme stejného konstruktory pro všechny instrukce při vytváření slovníku instrukcí v třídě `XmlParser`.