

## 1 Interpret

### Zpracování parametrů

Parametry jsou zpracovány pomocí třídy `program_arguments`. Třída má metodu `process_args`, ve které projde pole parametrů a parametr provede, v případě `--help`, případně ukládá soubory do svých atributů, v případě `--input` a `--source`.

### Čtení instrukcí

Instrukce jsou ze zvoleného vstupu postupně čteny a vytvářeny s případnými argumenty. V moment, kdy jsou zpracovány všechny instrukce na vstupu se jde na vykonání daných instrukcí.

### Instrukce

Všechny instrukce dědí z jedné třídy `instruction`. Tato třída má sdílený list, do kterého se při čtení dávají všechny instrukce. Dále má tato třída metodu `error_exit`, která při chybě vypíše chybovou hlášku s `opcode`m, `order`em a danou chybovou zprávou a poté ukončí program s odpovídající chybovou hodnotou.

Ze základní třídy `instruction` dědí čtyři třídy podle počtu argumentů instrukce, `no_args_instr`, `one_arg_instr`, `two_arg_instr`, `three_arg_instr`. Argumenty mají třídu `argument`.

Každá instrukce má svou vlastní třídu, kde implementuje vlastní metodu `execute`, jenž provede danou instrukci.

Provádění instrukcí poté probíhá pomocí `run` metody. Před použitím `run` se seřadí list instrukcí podle `order`. Poté se provede pomocí `execute` metody instrukce na indexu daném v `program_scopes` po provedení se toto číslo inkrementuje a opět se zavolá metoda `run`.

### Rámce

Interpret využívá třídu `program_scopes` jako zastřešení pro všechny rámce, zásobník, zásobník s indexem instrukce pro `RETURN` a číslo instrukce k provedení.

Jednotlivé rámce jsou instance třídy `scope`. Tato třída má typ, `GF`, `LF` nebo `TF`, a ukládání proměnných je implementováno pomocí `directory`, kde klíč je jméno proměnné (bez prefixu udávající její rámec) a hodnota je instance třídy `variable`.

Zacházení s proměnnými probíhá pomocí metod `def_var`, `get_var`, `set_var`. Podle prefixu se rozhodne, která další metoda bude volána.

## 2 Testovací skript

### Zpracování parametrů

Parametry jsou zpracovány pomocí třídy `program_arguments`. Třída funguje na podobném principu jako u interpretu, až na to, že namísto vlastní funkce pro zpracování parametrů se využívá funkce `getopt`.

### Testy

Testovací sada má třídu `test_set`, která obsahuje list, který se naplní instancemi třídy `test`. Poté se podle toho, jestli se testuje parse skript, interpret nebo obojí, provede testování.

### Výstup

Po projití všech testů se na standardní výstup vypíše HTML reprezentace výsledků ve formě dvou sloupců `Passed tests` a `Failed tests`, kde jsou jednotlivé testy, které prošly nebo neprošly.