

Krátký popis

Skript `interpret.py` načte XML reprezentaci programu a tento program s využitím vstupu dle parametrů příkazové řádky interpretuje a generuje výstup

Specifikace řešení

Projekt se skládá z řídicího souboru a souborů, které obsahují pomocné třídy:

- `interpret.py` - Nachází argumenty při spouštění programu, nachází `--input` a prochází XML strukturou ze `--source`.
- `Program.py` - Inicializuje hlavní objekt třídy `Frame` pro správu paměti. Prochází všechny instrukcí a inicializuje pro ně příslušné třídy z dokumentu `Operations.py` na základě rozcestníku. Připravuje pro operace jazyka IPPcode23 argumenty instrukcí ve správném tvaru.
- `Instruction.py` - Naplnění jednotlivých instrukcí kódem operací (`opcode`) a objektem pro argumenty (`arg1`, `arg2`, `arg3`). Obsahují informace o typu, rámci a hodnotě argumentu.
- `Operations.py` - Seznam operací jazyka IPPcode23 ve formě tříd (podrobný popis je uveden v komentářích, a je kopií specifikací operací v zadání).
- `Frame.py` - Třída pro vytváření objektu, reprezentujícího paměť programu. Obsahuje implementaci GF, TF a LF v podobě slovníků a zásobníku, a také metody pro práci s nimi: `set`, `get`, `push`, `pop`, `create`, `defI`.