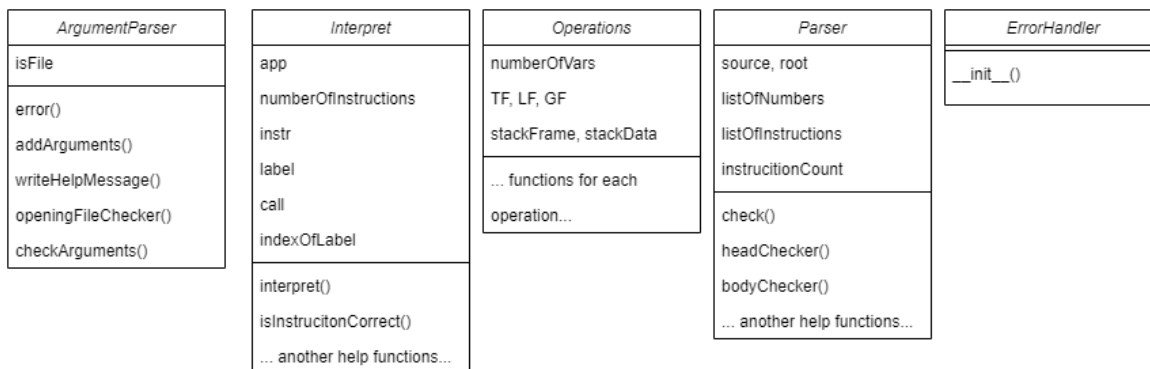


O programe

Program slúži na preklad vstupného súboru XML (jazyk IPPcode23) a jeho interpretáciu. Je implementovaný pomocou OOP, obsahuje 5 tried a jeden hlavný súbor `interpret.py`.

Triedy

- `ArgumentParser`: slúži na parsovanie argumentov. Je to prepísanie a navýšenie vytvorenej triedy, v prípade prepísania sa jedná o funkciu `error()`, v prípade pridania o pomocné funkcie potrebné na parsovanie argumentov.
- `ErrorHandler`: je vytvorený v prípade chyby. Pri tvorbe triedy sa hneď vypíše chybová hláška a program je ukončený
- `Interpret`: jadro programu, riadi postup v programe, volá operácie nad dátami. Funkcie, ktoré ovládajú chod programu a nachádzajú sa v tejto triede sú `interpret()` a `isConstructionCorrect()`, ktoré podrobnejšie popisujem nižšie.
- `Operations`: obsahuje operácie pre jednotlivé funkcie jazyka IPPcode23. Zásobníkové ukladanie prebieha pomocou `stackFrame` a `stackData`.
- `Parser`: parsuje XML súbor a vráti dva zoznamy s číslami inštrukcií a zoznam inštrukcií



Chod programu

Ako prvý je spustení súbor `interpret.py`, v ktorom prebieha riadenie programu. Do neho sú nainportované knižnice `sys`, `re`, `classes.Operations` a `classes.ErrorHandler`.

V prvom rade prebieha parsovanie argumenty pomocou triedy `ArgumentParser`, následne parsuje XML súbor pomocou triedy `Parser` a pokračuje interpretáciou pomocou triedy `Interpret` za pomoci `Operations`. V prípade, ak sa nevyskytuje žiadna chyba, je progrma ukončený s návratovým kódom 0 a výsledkom na `STDOUT`. V prípade chyby je vytvorená trieda `ErrorHandler`.

Funkcie `interpret()` a `isConstructionCorrect()`

`interpret()`: obsahuje cyklus `while` obsahujúci `switch`, ktorý roztriedí a volá jednotlivé operácie pre dané inštrukcie. Cyklus `while` prechádza inštrukciami, pokiaľ jeho iterácia nedosiahne čísla rovného počtu inštrukcií.

`isConstructionCorrect()`: taktiež obsahuje cyklus `while`, v ktorom sa zisťuje, či inštrukcie sú správne (majú dobrý počet argumentov) a taktiež či názvy, metódy a hodnoty symbolov sú korektné a správne. V prípade, že tomu tak nie je, je program ukončený pomocou triedy `ErrorHandler`.

Implementované rozšírenia

Implementoval som rozšírenia `FLOAT` a `STACK` a z časti `STATI`, ktoré som z časových dôvodov nedokončil.

`FLOAT`: podporuje čísla typu `float`

`STACK`: jednotlivé funkcie obsahujú parameter `saveToStack`, ktorý kontroluje, či bude podporované zásobníkové ukladanie.

Testovanie

Na testovanie som využil vlastné testy, ktoré sú aj prispôbené jednotlivým rozšíreniam. Pomocou testov som dosiahol hodnotu 492 správnych z 519 všetkých testov. Testy som vytváral ešte pred začatím implementácie programu, takže sa mi nepodarilo dosiahnuť splnenie všetkých testov.