

## Implementačná dokumentácia k 2. úlohe do IPP 2019/2020

Meno a priezvisko: Roman Fulla

Login: xfulla00

### Úvod

Úlohu som sa opäť pokúsil rozdeliť do viacerých súborov a rôznych objektov.

### interpret.py

Skontroluje parametre dané súboru, otvorí vstupné a výstupné súbory a pomocou knižnice `ElementTree` spracuje vstupnú XML reprezentáciu kódu. Následne presmeruje súbor so vstupmi na štandardný vstup (pre potreby vstavanej funkcie `input()`) a vytvorí objekt `program`, ktorý spustí. V prípade žiadosti vypíše štatistiky programu do štatistického súboru.

### program.py

Objekt `program`. Zo spracovanej XML reprezentácie si vytvorí zoznam inštrukcií, ktorí skontroluje a zoradí podľa parametru `order`. Dopredu si vytvorí slovník návěstí na ktoré môže skákať. Potom vykonáva inštrukciu po inštrukcii s prípadnými skokmi. Obsahuje zopár pomocných metód, ktoré mu napríklad umožnia nastaviť hodnotu premennej alebo ju získať. Automaticky získava a aktualizuje štatistiky o programe pomocou metód `__addinsts()` a `__addvars()`, ktoré následne môže na žiadosť interpretu vypísať.

### instruction.py

Obsahuje definíciu objektu inštrukcie. pri vytvorení vytvorí zoznam svojich parametrov, ktorý následne skontroluje. Obsahuje metódy, ktoré skontrolujú či funkcia naozaj obsahuje správne parametre. To zabezpečuje metóda `__argcheck()` volaná inými metódami, ktoré jej poskytnú aké parametre má konkrétna funkcia obsahovať.

### argument.py

Obsahuje definíciu objektu argumentu. Najskôr skontroluje správnosť argumentu a aj to, či sa jeho skutočný typ zhoduje s deklarovaným typom pomocou `regexu`. Obsahuje metódy pre prístup k svojim atribútom.

### exceptions.py

Obsahuje objekty výnimiek. Hlavná výnimka obsahuje metódu `last_words()`, ktorá vypíše správu o chybe a ukončí skript s korektnou návratovou hodnotou. Ostatné výnimky túto metódu dedia a menia návratový kód podľa potreby.

**test.php**

Z časových dôvodov som túto časť projektu neodovzdal.