

Dokumentace úlohy CLS C++ Classes v Python 3 do IPP 2015/2016

Jméno a příjmení: Jan Pavlica

Login: xpavli78

Zpracování parametrů

K zpracování parametrů jsem použil jednoduché porovnání části zadaného parametru. V případě shody jsem zbývající část parametru použil jako požadovanou hodnotu pro daný parametr. V případě zadání jiného než povoleného parametru je skript ukončen chybou. Nepovolené kombinace nebo opakování argumentů je ohlášeno booleovskými proměnnými.

Zpracování vstupního souboru

Vstupní soubor je nejprve načten jako celek a poté je rozdělen do jednotlivých logických celků, se kterými se dále pracuje. Prvně dochází k odstranění přebytečných bílých znaků. Dále následuje rozdělení po jednotlivých třídách (celé definice/deklarace). Tyto celky jsou dále zmenšeny jenom na samotná těla tříd a dále rozděleny, takže vzniká seznam metod, atributů a oprávnění

Zpracování informací o třídě

Třída `classes`

Tato třída slouží k uchování potřebných informací pro jednotlivé třídy. Obsahuje seznam tříd, ze kterých dědí, seznam tříd do kterých dědí, seznam atributů a seznam metod.

Každý atribut obsahuje informace o svém datovém typu, jménu, oprávnění v aktuální třídě, zda-li se jedná o statický atribut, jméno třídy ve které je definován a oprávnění v této původní třídě.

Každá metoda obsahuje informace o svém datovém typu, jménu, argumentech, oprávnění v aktuální třídě, zda-li se jedná o virtuální metodu, zda-li se jedná o statickou metodu jméno třídy ve které je definována a oprávnění v této původní třídě.

V třídě jsou vedeny dva seznamy pro atributy a dva seznamy pro metody. První obsahuje pouze prvky definované v dané třídě. Druhý poté obsahuje všechny prvky, tudíž i zděděné.

Získání jména, ověření definice a zjištění dědičnosti

Jako první se provede získání jména třídy. Tato třída je poté vložena do slovníku, kde jsou uchovány jednotlivé třídy. Jako klíč slouží jméno třídy, hodnotou je poté instance pro danou třídu. V případě, že se jedná pouze o deklaraci, tak se postupuje k další třídě, v opačném případě je pak zpracováno tělo třídy.

Získání zbývajících údajů

V případě, zpracování těla se postupně zpracovávají jednotlivé celky. Prakticky můžeme narazit jenom na tři případy: metoda, atribut, oprávnění. V případě, že se jedná o oprávnění, je nastaveno dané oprávnění, které určuje oprávnění další prvků. Pokud není specifikováno žádné oprávnění, uvažuje se oprávnění `private`.

Metoda a atribut jsou poté odlišeny regulárními výrazy. Regulárními výrazy je také kontrolován výskyt klíčových slov `static` a `using`. K získání informací o metodě slouží funkce `parse_method`, pro získání informací o atributu nebo argumentu metody pak `parse_atribut`. Při přidávání jednotlivých metod a atributů je kontrolováno zda-li se nejedná o redefinici v rámci jedné třídy.

Zpracování dědičnosti a kontrola konfliktů

Poté následuje zpracování dědičnosti. V případě, že daná třída dědí je zjištěno z jaké třídy a s jakým oprávněním dědí. S rodičovské třídy jsou poté postupně přidávány atributy a metody dle daného oprávnění. V této části také probíhá kontrola konfliktů. Ve skriptu je implementováno rozšíření `--conflicts`. Když je tento argument zadán, tak v případě nalezení konfliktu, zaznamenaný konflikt a s danými členy již dále nepracuje. Konflikty jsou uloženy ve slovníku, kde je klíčem jméno konfliktního členu a hodnotou poté seznam konfliktních členů.

Tvorba obsahu výstupního souboru

Strom dědičnosti

Pro tisk stromu dědičnosti bylo nutné uložit do každé třídy informaci o tom, které třídy z dané třídy dědí. Samotný tisk poté probíhá rekurzivním voláním funkce, která je jako první volána třídou, která z žádné jiné nedědí a poté dále rekurzivně volá třídy, které z ní dědí.

Detailní výpis

Při detailním výpisu jsou třídy tisknuty abecedně. V rámci každé třídy je poté volán tisk jednotlivých metod a atributů podle daného oprávnění.

Výsledek vyhledávání výrazu XPATH

Pro tyto účely byla použita knihovna `xm1`, která umožňuje vyhledávání XPATH výrazem. V případě, že je výsledkem vyhledávání element je tento element vložen jako podelement elementu `result`. Výsledný strom je poté převeden na řetězec. V případě, že je výsledkem vyhledávání řetězec, jsou jednotlivé řetězce tisknuty jdoucí abecedně po sobě. Jako poslední krok je poté úprava odřádkování a přidání hlavičky.