Dokumentace úlohy CSV: CSV2XML v Perlu do IPP 2011/2012
Jméno a příjmení: Pavel Frýz
Login: xfryzp00

Dokumentace k projektu IPP: CSV2XML v Perlu

1. Zadání

Úkolem projektu bylo implementovat převod CSV souboru do XML. Skript na vstupu přijímá soubor v CSV formátu (viz přepínač `--input`) a na výstup generuje vytvořený XML soubor (viz přepínač `--output`), tvar výsledného XML souboru lze konfigurovat pomocí zadaných přepínačů při spuštění programu.

2. Popis řešení

Zpracování parametrů

Pro zpracování parametrů je využita funkce `GetOptions` z modulu `Getopt::Long`, tím je zajištěno, že nebudou zadány chybné parametry. Dále se kontroluje jestli nebyla zadána špatná kombinace parametrů nebo špatné atributy u zadaných parametrů. Pokud nastane chyba při zpracování parametrů program končí s chybovým kódem 1, případně při zadání špatných atributů u parametrů `-r` a `-l` s chybovým kódem 30. Zadané parametry lze zkrátit, pokud zkrácením nedojde k nejednoznačnosti parametrů.

Načtení vstupního souboru

Pokud je zadán přepínač `--input=SOUBOR` program načítá vstupní CSV ze zadaného souboru, v opačném případě načítá CSV ze standardního vstupu. Poté co je vstup načten, přidají se nakonec načteného vstupu znaky CR a LF, a takto upravený vstup je předán modulu `Text::CSV_PP`, který rozdělí soubor na jednotlivé sloupce a řádky. Před použitím modulu se nastaví separátor sloupců, podle parametru `-s` (identifikátor TAB pro tabulátor nebo libovolný znak, mimo ", CR a LF) a separátor řádků na `CRLF`. Mimo to se ještě nastavují atributy `binary` a `verbatim`. Vstup je poté rozdělen pomocí funkce `getline_all`, která ho načte do dvourozměrného pole. Jestliže vstup neobsahoval validní data, skript skončí s návratovým kódem 4 (s parametrem `validate` s kódem 39).

Generování výsledného XML

Pokud je zadán přepínač `--output=SOUBOR` program vypíše výsledný XML dokument do zadaného souboru, jinak ho vypíše na standardní výstup. Pro výpis souboru je použit modul `XML::Writer`, který nahrazuje některé nevalidní znaky XML dokumentu. Pro nahrazení nevalidních znaků v názvu tagů slouží funkce `replaceInvChar`, která nahradní nevalidní znaky pomlčkou. Pokud není zadán parametr `-n`, tak se do výsledného dokumentu vloží XML hlavička, která obsahuje verzi XML a použité kódování. Po vložení hlavičky se při použití přepínače `-r` přidá tag obalující výsledné XML. Následně se prochází pole, do kterého byl načten vstupní soubor a pro jednotlivé řádky se generují tagy, mezi které se vkládají tagy obalující sloupce. Při nezadání přepínače `-l` je implicitní jméno tagů obalující řádky row, jinak se použije jméno z paramerů. Při zadání parametru `-l` je možné zadat také parametr `-i`, který k tagu řádku přidá atribut index s číslem řádku, číslo řádku může být specifikováno pomocí přepínače `--start`. Názvy tagů obalující sloupce lze změnit přepínačem `-h`, kdy se názvy odvodí z prvního řádku CSV souboru, implicitně jsou sloupce označeny colX, kde X je pořadí sloupce.

V průběhu generování se kontroluje jestli počet sloupců právě zpracovávaného řádku je stejný, jako počet sloupců prvního řádku. Pokud je počet sloupců různý a program nebyl spuštěn s přepínačem `-e|--error-recovery`, program je ukončen s návratovým kódem 32 (s parametrem `validate` s kódem 39). Při zadání parametru `-e` se chybějící sloupce doplňují hodnotou přepínače `--missing-value` nebo prázdným polem a přebytečné sloupce se ignorují. Parametrem `—all-columns` se docílí vypsání i přebytečných sloupců.

3. Rozšíření

V programu jsou implementována rozšíření PAD a VLC. Rozšíření PAD přidává nový přepínač `--padding`, který zajistí doplnění dostatečného počtu nul zleva u čítačů sloupců i řádku tak, aby byl počet číslic u uvedených čítačů stejný. Pokud je tento přepínač zadán tak se z počtu řádků pole s načteným vstupem odvodí počet nul pro index řádků. Stejná operace také probíhá pro počet sloupců jednotlivých řádků. Rozšíření VLC přidává přepínač `--validate`, zkontroluje, jestli je vstupní soubor validní. Kontrola probíhá na dvou místech, první provede

modul `Text::CSV_PP` a druhá se provádí při kontrole počtu sloupců.

4.Závěr

Program byl testován na dodaných a mnou vytvořených testech. Během testování bylo odhaleno několik chyb, které byly následně opraveny. Testování probíhalo na počítači s operačním systémemu Ubuntu a školním serveru Merlin s operačním systémem CentOS. Výsledný program dodržuje formát vstupních a výstupních dat, díky tomu je možné využití ve skriptech nebo spolupráce s jinými programi.