

## **Zadání**

Úkolem bylo vytvořit skript v jazyce PHP5, který provede konverzi vstupu(vstupního souboru) ve formátu JSON do XML. Celé zadání jsem rozdělil na jednotlivé úlohy, které jsem pak implementoval ve svém skriptu.

## **Implementace**

Mnou navržený program rozkládá zadaný problém na podproblémy, jako výpis chyb, zpracování parametrů, ověření validatnosti řetězce. Podproblémy jsou řešené pomocí podprogramů, např. podprogram `func_error()` v případě chyby ukončí skript s chybovým kódem a výpisem chybového hlášení.

## **Zpracování parametrů**

Zpracování parametrů zajišťuje interní funkce `getopt()`, která rozdělila pole řetězců na jednotlivé části. Pak tyto části byly uloženy do příslušných proměnných, se kterými budou pracovat další podprogramy. Tak se program stará o to, aby byl dodržen správný formát parametrů(--start se nemůže vyskytnout bez přepínače --index-items, --help nemůže být v kombinaci s jinými parametry).

## **Zpracování souboru**

Analýza souboru se skládá z několika částí. Nejdříve proběhne načtení vstupního JSON souboru(může být zadán ze standardního vstupu nebo relativní/absolutní cestou). Pak jeho transformace na formát PHP, pomocí interní funkce `json_decode()`. Pak skript posílá data do podprogramu JSON2XML, který provede převod.

## **Převod**

Podprogram JSON2XML se stará o převod PHP pole do XML formátu. Postupně prochází a zpracovává jednotlivé prvky, buď pole nebo objekty. V případě, že prvkem je další objekt nebo pole, podprogram se zanoří do rekurze. Jestli prvkem řetězec, číslo nebo literal (true,false, null) dojde k volání funkce `change_string()`. Ten podprogram ověří má-li v sobě vstupní řetězec nevhodné znaky a modifikuje jejich podle nastavených přepínačů. Na výstupu této funkce je řetězec převedený do XML formátu, který pak se vypíše na standardní výstup nebo se zapíše do souboru.

## **Problémy**

Největším problémem při implementaci projektu bylo převedení php řetězce do XML formátu. Musel jsem se postarat o správný formát výpisu (počet mezer, obalující elementy, xml hlavička, invalidní znaky a jejich transformace a t.d.).

## **Zavěr**

Skript byl otestován testy zadanými k projektu. Testování jsem provedl na školním serveru Merlin. Všechny testy dopadly úspěšně.