

## Implementačná dokumentácia k prvej úlohe do IPP 2020/2021

Meno a priezvisko : Emma Krompaščíková

Login : xkromp00

Kód som začala tým, že som si pre každú inštrukciu vytvorila regulárny výraz. Pre každý literál som mala vytvorený vlastný regulárny výraz. Tie som následne vkladala so ostatných regulárnych výrazov, na mieste, kde podľa zadania mali patriť. Každý regulárny výraz som obohatila o možnosť komentáru. Počítala som aj s možnosťou prázdneho riadku, alebo komentáru na prázdnom riadku, a preto som vytvorila regulárny výraz, ktorý zahŕňal obe možnosti. Vo veľkom while cykle načítavam a následne pomocou `preg_match` porovnávam jednotlivé riadky s regexami. Používam kontrolnú premennú, ktorej obsah zodpovedá počtu argumentov inštrukcie. Rovako mám aj kontrolnú premennú, ktorá obsahuje regulárny výraz, na ošetrovanie chyby číslo 22. Pokiaľ sa prvé slovo v riadku nezhoduje s jedným zo slov v regexe, vypisujem chybovú hlášku (komentáre, samozrejme, nerátam). Na vypisovanie XML používam knižnicu `xmlwriter`. Podľa počtu argumentov v kontrolnej premennej sa vo funkcii `if` rozhodujem, koľko typov bude vypísaných na výstup. Keď sa funkcia `if` nezhodne s počtom argumentov a rovnako, pokiaľ to nie je ani komentár, tak na chybový výstup vypisujem chybovú hlášku a exitujem číslo 23. Pri kontrole hlavičky hľadám regex, ktorý obsahuje presné znenie hlavičky, je ošetrovaný o biele znaky a komentáre. Pokiaľ hlavičku nenájdem, vypisujem chybovú hlášku a exitujem s číslom 21. Pri vypisovaní xml kódu mám ošetrované pomocou `strtoupper`, že opcode bude stále písaný veľkými písmenami. Pod regulárnymi výrazami na začiatku kódu mám dve funkcie. Jedna vracia typ argumentu, a druhá je potrebná pre správne vypísanie xml výstupu. Echo na vypísanie celého xml výstupu mám až na takmer úplnom konci kódu. Tým pádom sa xml kód vypíše naraz po prejení celého vstupného súboru, a nie počas.