

1. Stručný popis činnosti skriptu

Účelem skriptu je vyhledat v adresářové struktuře soubory programovacího jazyka C s příponou .h a zanalyzovat jejich obsah. Pro tyto činnosti skript využívá funkce pro zpracování parametrů příkazové řádky, zpracování regulárních výrazů, rekurzivní procházení adresářové struktury a nástroje pro tvorbu XML dokumentů.

2. Popis jednotlivých částí skriptu

Skript je rozdělen na několik částí, které jsou blíže popsány níže.

2.1 Zpracování parametrů

Zpracování parametrů je realizováno pomocí funkce `getopts`. Součástí zpracování parametrů je i testování, zda některý z parametrů nebyl zadán vícekrát nebo zda nebyl zadán parametr, který skript nezná.

2.2 Vyhledání hlavičkových souborů

Vyhledání hlavičkových souborů je realizováno pomocí funkce `RegexIterator`, která pomocí regulárního výrazu v kombinaci s funkcí `RecursiveDirectoryIterator` rekurzivně vyhledá v dané adresářové struktuře všechny soubory s příponou .h.

2.3 Analýza hlavičkových souborů

Analýza hlavičkových souborů vyhledá pouze samotné deklarace funkcí a probíhá v cykle. Nejdříve je obsah každého hlavičkového souboru načten pomocí funkce `file_get_contents`. Následně jsou s načteného obsahu odstraněny všechny nadbytečné úseky (komentáře, makra), a všechny bílé znaky jsou nahrazeny za mezeru. Náhrada bílých znaků a odstranění nadbytečných souborů probíhá pomocí několikanásobného volání funkce `preg_replace` se správním regulárním výrazem.

Následně je načtený obsah analyzován pomocí funkce `preg_match`, která pomocí regulárního výrazu vyhledá všechny deklarace funkcí v načteném obsahu. Hlavičkový soubor včetně jeho nalezených deklarací funkcí je uložen do pomocného dvourozměrného pole, kde první rozměr tvoří hlavičkový soubor a druhý rozměr tvoří deklarace funkcí daného hlavičkového souboru. Pokud hlavičkový soubor neobsahuje žádnou deklaraci funkce, tak se do pomocného pole neukládá.

2.4 Analýza deklarací funkcí

Analýza deklarací funkcí vyjme s každé deklarace funkce informace (název funkce, návratový typ, počet parametrů, ...), které jsou zapotřebí pro tvorbu XML dokumentu. Analýza probíhá souběžně s tvorbou XML dokumentu v jednom cyklu. Vyjmutí názvu funkce a návratového typu funkce probíhá pomocí funkce `preg_match`. Pokud je aktivní přepínač `--no-duplicates` tak se název funkce ukládá do pomocného pole. Parametry funkce jsou uloženy do pomocného pole pomocí funkce `preg_match_all`. Součástí analýzy je také test, zda má funkce proměnný počet parametrů, který probíhá pomocí funkce `preg_match`.

2.5 Tvorba XML dokumentu

Vytváření XML dokumentu probíhá pomocí nástroje `DOMDocument`. Nejdříve jsou do XML dokumentu vloženy položky `functions` a `dir`.

Následně proběhne analýza deklarací funkcí, která vyjme s funkcí potřebné informace.

Po analýze následuje vytvoření elementu `function`, které může být ovlivněno některými z parametrů příkazové řádky (`--no-duplicates`, `--no-inline`, `--max-par`). Následně se element `function` vyplní potřebnými atributy (`name`, `varargs`, `retype`). Pokud je aktivní přepínač `--remove-whitespaces` tak jsou s atributu `retype` odstraněny všechny přebytečné mezery pomocí funkce `preg_replace`.

Následně se vytvoří element `params`, pokud je počet parametrů větší než 0. Před zápisem atributů elementu `params` se parametry upraví (odeberou se přebytečné mezery) pomocí funkce `preg_replace`.

Po vytvoření všech elementů a jejich atributů se upraví odsazení XML dokumentu pomocí funkce `preg_replace_callback` podle toho, zda byl zadán parametr `--pretty-xml`.

Na závěr se vypíše XML dokument na standardní výstup, nebo do XML souboru pomocí funkce `file_put_contents` podle toho, zda byl zadán parametr `--output`.