Task documentation CHA: C Header Analysis in PHP 5 for IPP 2016/2017
Name and surname: Hoang Duong Nguyen
Login: xnguye14

The main goal of the script is to analyze all .h files in the directory and subdirectories. Script analyzes all functions in each .h file and writes them in XML format as specific elements. Some functions for processing directories, working with strings and creating XML elements, which are supported by PHP language, are used in this script.

Executing parameters:

Parameters of the script are read and saved to the defined array by PHP function ***getopt***. Script tests if some parameters are duplicated, invalid or are put in the wrong place(for example parameter *--help* must be in the first place and must be the only one parameter) and prints error message in standard error output and the specific value of the error is returned.

Analyzing files and create XML file:

Depending on the parameter --input, script will analyze the actual directory or the given directory or the specific file. Script finds all the .h files in the current directory and saves them to the array. It is the job of created function **input**, which contains PHP functions for processing all directories and subdirectories like ***RecursiveDirectoryIterator*** and ***RecursiveIteratorIterator***. Script opens the files one by one and all the functions in these files are scanned and saved to the array by PHP functions ***preg_match_all and preg_replace***. These function work with regex expressions which is very effective and it saves a lot of works. Script scans all the functions which are found and separates their parameters in form of strings.

Xml file is created by ***DomDocument*** class. The XML file contains header and 3 types of elements: functions, function and param. Element functions is the root and it has a name of the directory which is explored. Element function is a child of functions. This element contains information about each found function in the file such as name of the file in which this function is defined, name of the function, variable number of arguments and return type. Element param is a child of function and it contains information about each parameter of each function: serial number and type of the parameter.

In the end XML file will be printed in standard output or in the given file by parameter --output if this parameter is set. Depending on the parameters of the script, script will skip the functions which are defined as *inline* in the file or which are duplicated, script can skip functions which contain more than number n which is in parameter *--max-par=n,* script can also delete extra whitespaces in the return type of functions and parameters. XML file can be printed as one line or in separated line with specific indenting. It depends on parameter *--pretty-xml*.
The problem how to print XML in one line is solved by PHP function preg_replace by deleting all the new line character (replacing all the new line character with nothing). If *--pretty-xml* is set and if it contains *k* (integer) then each *<function>* element in XML file will be in new line and will be indented by *k* whitespaces or by default number if *k* is not set. For element *<param>* is *k* or default number twice as much. I solved this problem with for cycle with *k* or default number iterations, in each iteration the function ***preg_replace*** is used to replace *<function* and *</function* with _*<function* and _*</function* (_ is whitespace), the same with <param and </param in for cycle with twice iterations. This is not the best solution and it doesn't look good but due to lack of time this is the only solution which came out from my mind.