

Documentation of the implementation of the 1<sup>st</sup> IPP task.

Name and surname: Andrea Chimenti

Login: xchime00

## Command line arguments handling

The command line arguments are handled very simply in the main part of the script. The only supported argument is `--help` (also `-h`), which instructs the script to print on `stdin` a simple description of the script functionality and end with error code 0.

## IPPCODE20 analysis

The analysis of IPPCODE20 source code is handled by `Parser` class, which encapsulates the whole functionality of the script. An instance of the class, which is then used to generate the XML representation of the IPPCODE20 source code, is created at the beginning of the program. The method `generate_xml()` examines whether a valid header is present, then reads the input line by line. The value of the operation code on each line is used as a key in the `$iia` (instruction info array), which determines what parameters are expected on the rest of the line. The parameters are checked with regular expressions. This operation should cover most of the lexical and syntactic analysis. The line is then inspected again and elements of the XML DOM are created.

## XML generation

The `Parser` class uses the `DOMDocument` library to generate the XML output. An instance of the `DOMDocument` is created in the constructor of the `Parser` class. At the beginning of the generation, the version and encoding attributes are specified in the xml header and a program element is created. Every time the script processes a line, the proper operation and argument elements are created and appended to the xml dom document. At the end of the script, the content is printed out.