

# COP701: Assignment #1

July 26, 2023

## 1 Markdown to HTML converter

### 1.1 Problem Statement

This is your first assignment in the COP 701 course. In this assignment your main objective is to convert a Markdown document to an equivalent HTML document. In pursuance of this objective, you will have to write a Markdown to HTML parser from scratch.

The features of markdown which you all need to consider are:

- headings - level 1 to 6
- paragraphs
- line breaks
- bold and italics emphasis
- ordered and unordered lists
- images
- URLs(hyperlinks)
- tables

You can consider other features for extra credit.

### 1.2 Workflow and subtasks

The entire assignment can be divided into the following sub-tasks:

- Learn about markdown and HTML in brief.
- Write a lexer i.e to do a lexical analysis of your markdown document and generate a string of tokens. Programs that you can use: flex, jflex
- Do not use any available libraries to parse the markdown.

- Parse the sequence of tokens using parser such as yacc, CUP, ANTLR, bison (C++ or Java)
- Generate an AST(Abstract Syntax Tree) of your markdown document. [link](#)
- Map it to an equivalent AST of HTML.
- Generate the equivalent HTML code which can be rendered by a browser.

### 1.3 Links to important resources

- Know about Flex tool and performing lexical analysis using that.
- Some more resource regarding Flex.
- Introduction to an Abstract Syntax Tree.
- Bison parser generator.
- YACC parser generator.

## 2 Logistics

- You are free to code in **any** programming language except python. Eg: C/C++/Java/Rust/Go
- The **deadline** for this assignment is **27/08/2023 at 11:59 PM**. It is a hard deadline and will not be extended.
- This is an individual assignment (30 Marks)
- You need to create a private git repository either on <https://git.iitd.ac.in> or github. Git commit history will be checked during evaluation.
- References for coding style can be found on the course webpage. Eg: C/C++, java.
- You need to write unit tests for your code.
- All the modules of your code need to be documented. Eg: Javadoc, Godoc.
- Use a Makefile for the project build by defining targets, dependencies, and build commands.
- You are not expected to use any external libraries to directly convert to HTML.
- ANY form of **plagiarism** will not be tolerated.

- Also, create a **run.sh** file where the first argument will be the name of the markdown file and the second argument will be the name of the output HTML file. We will run the command `./run.sh input.md output.html` during the evaluation/demo.
- Submission will be made on Moodle. You need to submit all your code (parser, translator) and a pdf format report. Compress all these in a tar file with the name `<entry_number> .tar` and upload on Moodle.
- You will be graded on the output of your code, the coding style and your viva/presentation.
- Marks distribution: Coding style - 25% , Demo - 75%
- We will be testing on hidden test cases during the demos.
- Any doubts regarding the course/assignment should be asked on Piazza.