

# Assignment 3

## Deadline

11:55 PM on 12th March, 2018

## Problem

Write a Parser for the source language that you have chosen. The output of the parser must print the ***rightmost derivation*** of the input program in a nicely formatted *html* file. The non-terminal being expanded must be marked in a different color/bold-faced/underlined etc. to make it easy to follow the derivation. One of the ways to build the assignment is to first generate the rightmost derivation in reverse and then use the “**tac**” command to reverse the file.

## Details

- Your implementation should read the source filename as its first command-line parameter; it should produce its output as an html file. For example, for an input program “abc.c”, it should produce a dot file “abc.html”.
- You must use **your** implementation of the lexer done as a part of Assignment 1. You are allowed to do minor modifications over your assignment 1 submission.
- You must use a parser generator like Yacc, Bison etc.
- The tool should be robust; syntactically incorrect programs must be duly reported.
- You have to submit a zipped folder (name the folder “asgn3”) with:
  - the source of the implementation (in a folder called “src” within “asgn3”;
  - a Makefile to build the implementation (it should generate an executable called “parser” in the folder “asgn3/bin”;
  - a set of at least 5 test cases that you have used to check your implementation (in a folder “asgn3/test”);
  - a README file with a brief description for building and running it (within “asgn3”).

Binaries should NOT be part of the submission. Clean the folder of all object and executable files before submission.

- We will apply the following set of commands to build and run your implementation; make sure that your implementation works correctly with these sequence of commands:
  - cd asgn3
  - make
  - bin/parser test/test1.c (to execute the first test-case file test1.c)
  - firefox test1.html