Computer - Aided Circuit Design and Analysis 2022 Spring

# Project 2

# LU Solver

## Deadline: 2022/5/27 23:59

## Project Introduction

In this project, you are going to write a **LU** solver that can solve *any linear resistive circuit with no controlled sources* using modified nodal analysis (MNA). You can use **Project 1** to do the parser and stamping functions.

## LU Solver

The linear solver should use Gauss's method for **LU**-factorization, partial (row) pivoting, and using an in-place computation so that **L** and **U** simply over-write the system matrix. **Partial pivoting** (using row-exchange) should be performed to find the best pivot for accuracy, and not only to avoid a zero pivot. You can ignore issues of sparsity.

Your implementation should be general, in the sense that it should accept any linear circuit description consisting of any combination of linear resistors and independent voltage and current sources. Use your code to solve the test circuit given in **Fig. 1** and **input_circuitDiscription.txt**, where the 10Ω and 50Ωresistors are required to be in group 2. The correct solution is V(4)=1.9888V, V(8)=1V, V(3)=2.00897V, V(2)=1.80879, V(6)=1.98814, V(5)=2V, V(1)=1.88527V, V(7)=3.98814V, I(R8)=198.88mA, I(R3)=3.82mA, I(V3)=0A, I(V2)=-199.88mA, and I(V1)=-198.88mA.

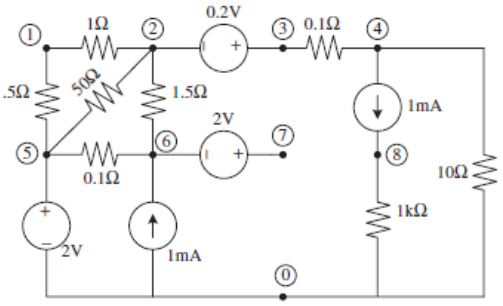


Fig. 1 A reference circuit of input_circuitDiscription.txt

## Output Format

An example of output is shown in **output_example.txt**. You should follow the format of this file.

## Language

C/C++

## Platform

Linux (Please make sure your code is available on linux server).

## Warning

Several circuit description files will be put into your program for testing and be scored.

## Naming, Programming, and Command rules

You should output the file of the results as the format shown in above examples.
Your program should take the command-line arguments as follows:
``**./Project2 [input] [output]**''
(example: **./Project2 input_circuitDiscription.txt output_example.txt**)

## Submission

Please upload the following materials in a .zip file (e.g. **Project_2_StudentID.zip**) to E3 by the deadline, specifying your student ID in the subject field.
(1) S Source code (.cpp/.c, .h, or Makefile).
(2) Executable binary.
(3) A Readme file (Information of how to compile/execute your codes/program.)

## Grading Policy

(1) **Plagiarism is not allowed.**
(2) **Correctness: 80%, Runtime: 20%.** If the output files are correct, you will get at least 80 points. Note that several hidden cases will be tested on your program.