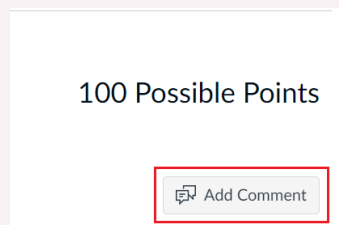


## CSC 371 – Finite Automata

### Project 2 (Due date: 12/10)

#### Instructions:

- (1) Deadline of the submission is 11:59 PM on 12/10. Late assignments will be accepted within 24 hours after the deadline and 15% points loss penalty will be applied. The submission page on Blackboard will be closed after 11:59 PM on 12/11, and no further assignments will be accepted after that time point.
- (2) For this project, you have the option to work in groups, with **each group limited to a maximum of two students**. However, it is important to note that the intention behind allowing group work is not solely to lighten individual workloads. Instead, it aims to provide you with an opportunity to enhance your collaboration skills and learn from one another. Make sure that the project's workload is distributed equally among group members. Both group members will receive the same grade. Therefore, only one submission needs to be uploaded by each group. Write down the names of all group members in the designated "Add Comments" section of the submission page on Blackboard, as shown below:



- (3) **Adhere to all the specified requirements without making any modifications.** Please ensure strict compliance throughout the project. For instance, one specific requirement is to read a text file. Consequently, your program must be capable of successfully reading a text file. Deviating from this requirement, such as prompting the user to input the transition table of an NFA instead, will result in the loss of all points allocated for this particular component.
- (4) You can use any programming language, however, Java is preferred. **After you finish, run your program several times to check the correctness. Make screenshots of at least three executions including input and output (sample input and output can be found in the problem description below), put the screenshots in the project folder, and then export the entire project as a zip file and upload that single zip file to Blackboard. I highly recommend that you design some other testing files yourself. Your submission should be one and only one zip file.**

**If you do not follow the above instructions, at least 10% points loss penalty will be applied.**

- (5) **All work turned in be the students' own work. Plagiarism and cheating will not be tolerated. Please refer to our syllabus for more information about plagiarism and cheating.**

**I consider plagiarism and cheating to be serious offenses. I highly recommend that you do not refer to any code from anybody. Reference to the following websites or similar websites is strictly prohibited. I have seen students from previous semesters that received 0 because of copying the code from online.**

● <https://github.com/aliyazdi75/Simplifying-CFG/tree/master/src>

● <https://www.chegg.com/homework-help/questions-and-answers/help-java-code-leave-notes-q88425273>

\*\*\*\*\*

## Project Problem: Simplification of Context-Free Grammars

We learned three context-free grammar (CFG) simplification techniques in our classes: removing  $\epsilon$ -rules, unit-rules, and useless rules. In this project, read a CFG from a txt file, simplify it by **removing  $\epsilon$ -rules and useless rules**, and print out the simplified equivalent CFG. No need to care or remove unit-rules.

For example, given the following CFG in a txt file (0 denotes empty string and “-” denotes arrow head “ $\rightarrow$ ”):

```
S-aA|aBB
A-aaA|0
B-bB|bbC
C-B
```

After processing, your program should print out the following simplified equivalent CFG:

```
S-aA|a
A-aaA|aa
```

The following are a few more examples you can use to test your program (I highly recommend that you come up with more examples to make sure the correctness of your program):

(1) S-AaBaCbD

```
A-0|a
B-0|b
C-0|c
D-0|d
```

For the above CFG, the print out should be

```
S-aab|Aaab|aBab|AaBab|aaCb|AaaCb|aBaCb|AaBaCb|aabD|AaabD|aBabD|AaBabD|aaCbD|AaaCbD|aBaCbD|AaBaCbD
A-a
B-b
C-c
D-d
```

(2) S-AaB|aaB

```
A-0
B-baA|0
```

For the above CFG, the print out should be

```
S-aB|aaB|a|aa
B-ba
```

(3) S-ASA|aB

```
A-B|S
B-b|0
```

For the above CFG, the print out should be

```
S-ASA|aB|a|SA|AS
```

A-B|S

B-b

Note that your program must be able to work for any CFG, not only just for the above given examples.

\*\*\*\*\***END**\*\*\*\*\*