

Programming assignment 6 (40 points)

[Start Assignment](#)

Due Tuesday by 11:59pm **Points** 40 **Submitting** a file upload **File Types** tar
Available until May 19 at 11:59pm

Simulating basic digital logic circuits (40 points with opportunity for 40 points extra credit)

Prof. Yipeng Huang

Rutgers University

2021 Spring

0. Introduction and setup

Learning goals

You will gain familiarity with digital logic circuits that underpin all computers. This programming assignment will further solidify your ability to use the C programming language to read files, store data in dynamically sized data structures, and perform Boolean operations. This programming assignment will help you learn about Boolean algebra and truth tables.



Resources

Get started early! You can post questions to the class Piazza (please avoid sending emails to the TAs and instructor).

- You can discuss the assignment with your classmates, but you cannot copy code from your classmates.
- You can use the internet to research the mechanics of the C programming language, the documentation for useful C functions, how Boolean algebra works, and truth tables, but you cannot

use websites that offer ready-made answers specific to this assignment and this class. Consider citing any sources that you use in your research; such a citation can be as simple as providing a link to a webpage in a code comment.

- We will be checking the assignments for identical and/or plagiarized code using automated tools.
- It is not acceptable to share or post any course materials online without explicit permission from the instructor.
- <https://www.cs.rutgers.edu/academics/undergraduate/academic-integrity-policy>.
(<https://www.cs.rutgers.edu/academics/undergraduate/academic-integrity-policy>)
- <https://www.cs.rutgers.edu/academics/undergraduate/academic-integrity-policy/programming-assignments>
(<https://www.cs.rutgers.edu/academics/undergraduate/academic-integrity-policy/programming-assignments>)
- The instruction staff takes the Rutgers University Academic Integrity Policy seriously. Please review it at: <https://nbprovost.rutgers.edu/academic-integrity-students>
(<https://nbprovost.rutgers.edu/academic-integrity-students>). We will report all violations.

Setup

Access the programming assignment files for this assignment by pulling new files from the class Github repository, 2021_0s_211:

```
git pull
```

If you do not have this directory already, clone the repository:

```
git clone https://github.com/yipenghuang0302/2021_0s_211.git
```

The files for this assignment are in the directory 2021_0s_211/pa6/.



basicGates: Simulating basic digital logic gates

(medium) (40 points)

Your task in this assignment is to read a text file that describes a digital logic circuit and output the truth table for the circuit.

Input format

Your program should take a single command line argument specifying the path to an input file. The input file is a digital circuit description that lists how many input and output variables are in the circuit, the names of the input and output variables, and digital logic gates that describe how the output variables relate to the input variables.

- The first line of the file will look like this: "INPUTVAR 2 input0 input1". The number N indicates the number of Boolean input variables to the digital logic circuit. In this case N=2. The rest of the line lists the names of the input variables, separated by spaces.
- The second line of the file will look like this: "OUTPUTVAR 1 output0". The number M indicates the number of Boolean output variables from the digital logic circuit. In this case M=1. The rest of line lists the names of the output variables, separated by spaces.
- The remainder of the file will have lines describing digital logic gates. The gates have either a single input or two inputs:
 - Single input gates include the "NOT" gate. A line describing such a gate will look like this: "NOT input0 temp0". This means the Boolean value of temp0 = ! input0. The intermediate Boolean variable temp0 is a variable in the circuit that is neither part of the inputs nor outputs of the digital circuit, but you need to keep track of temp0's value to calculate the circuit's truth table.
 - Double input gates include the "AND", "NAND", "OR", "NOR", "XOR", and "XNOR" gates. A line describing such a gate will like this: "XNOR input1 temp0 output0". This means the Boolean value of output0 = input1 XNOR temp0.

Example input files are in the tests/ directory.

Output format

Expected outputs from your program for each test case are in the answers/ directory. You should print lines that represent rows of the truth table for the digital circuit.

- For a digital circuit with N input variables and M output variables, each row should have N+M columns of binary digits (0 and 1). The first N columns correspond to the values of the input variables listed in the INPUTVAR input file line. The last M columns correspond to the values of the output variables listed in the OUTPUTVAR input file line. Do not print the Boolean values for any temporary variables that you kept track of.
- For a digital circuit with N input variables, there should be 2^N rows listing all the permutations of the input variable. The sequencing of the rows doesn't matter. The autograder will recognize different orderings for the rows as equivalent, and they would all be correct assuming that you list all 2^N possible input combinations.

How to compile, run, and test your code

- Instructions from Programming Assignment 1, 2, 3, and 5 carry over.
- The autograder.py scripts for this assignment only work with Python version 3 on the ilab machines. This means that the correct way to invoke the autograder script is through either of these two commands:

```
./autograder.py
```

or

```
python3 autograder.py
```

2. multiInput: Simulating multiple-input digital logic gates (hard) (40 points extra credit)

Your task in this section is modify your simulator from part 1 basicGates so that it handles basic digital logic gates that can have more than 2 inputs.

Input format

The input format for the previous part, basicGates, carries over, with these important differences:

- Instead of double input gates, the input files now have multi-input gates that include the "AND", "NAND", "OR", and "NOR" gates. A line describing such a gate will look like this: "AND 3 input1 input0 input1 temp0". The second entry on the line indicates that this AND gate takes three inputs, which are immediately listed before listing the single output variable. This means the Boolean value of $\text{temp0} = \text{input1 AND input0 AND input1}$.
- The input files will no longer have "XOR" and "XNOR" gates as these gates do not have a strict definition for multiple inputs.



Output format

The output format for the previous part, basicGates, carries over. Expected outputs from your program for each test case are in the answers/ directory.

How to submit

From the pa6/ directory, you can run this command to check on the outputs of our autograder script.

```
./assignment_autograder
```

or

```
python3 assignment_autograder.py
```

When you are ready to submit, from the 2021_0s_211/ directory where you see the pa6/ directory, run this command:

```
tar cvf pa6.tar pa6/
```

Upload the file pa6.tar here on Canvas.

By submitting your assignment, you are agreeing to the Rutgers Honor Pledge: “On my honor, I have neither received nor given any unauthorized assistance on this assignment.”

We will not be accepting late assignments. The Canvas submission site will close to enforce the deadline, so be sure to submit early.

If anything is not clear, reach out to your classmates and the instructors on the class Piazza!

