ECE 374 B ✧ Spring 2023
Homework 1

- Groups of up to three people can submit joint solutions. Each problem should be submitted by exactly one person, and the beginning of the homework should clearly state the Gradescope names and email addresses of each group member. In addition, whoever submits the homework must tell Gradescope who their other group members are.

- **Submit your solutions electronically on the course Gradescope site as PDF files.** If you plan to typeset your solutions, please use the LaTeX solution template on the course web site. If you must submit scanned handwritten solutions, please use a black pen on blank white paper and a high-quality scanner app (or an actual scanner, not just a phone camera).

☞ Some important course policies ☜

- **You may use any source at your disposal**—paper, electronic, or human—but you ***must*** cite ***every*** source that you use, and you ***must*** write everything yourself in your own words. See the academic integrity policies on the course web site for more details.

- **Avoid the Three Deadly Sins!** Any homework or exam solution that breaks any of the following rules will be given an ***automatic zero***, unless the solution is otherwise perfect. Yes, we really mean it. We're not trying to be scary or petty (Honest!), but we do want to break a few common bad habits that seriously impede mastery of the course material.

  - Always give complete solutions, not just examples.
  - Always declare all your variables, in English. In particular, always describe the specific problem your algorithm is supposed to solve.
  - Never use weak induction.

See the course web site for more information.

If you have any questions about these policies,
please don't hesitate to ask in class, in office hours, or on Piazza.

1. Give the recursive definition of the following languages. For both of these you should concisely explain why your solution is correct.

   (a) A language L_A that contains all palindrome strings using some arbitrary alphabet Σ.

   (b) A language L_B that does not contain either three 0's or three 1's in a row. E.g., $001101 \in L_B$ but 10001 is not in L_B.

2. For each of the following problems:

   i. Formulate the problem as a *regular* language (give an example of the problem instances and how they are encoded, you don't have to write every problem instance).

   ii. Describe the regular expression that describes the expression

   Note that how you encode the language matters for the regular expression you end up with.

   a Checking whether (or not) a number is divisible by 4). You are given a binary number and need to output if this number is divisible by 4.

   b The game of TicTacToe. You are given a completed tic-tac-toe board and you need to determine who won. (this won't have a clean regular expression. Just define some encoding and describe how you would build the expression, you don't need to write the whole expression out.) Hint: think about how many games of TicTacToe there are.

3. **Regular expressions I:** For each of the following languages over the alphabet $\{0, 1\}$, give a regular expression that describes that language, and briefly argue why your expression is correct.

   (a) All strings that end in 1011.

   (b) All strings except 11.

   (c) All strings that contain 101 or 010 as a substring.

   (d) All strings that contain 111 and 000 as a subsequence (the resulting expression is long – describe how you got your expression, instead of writing it out explicitly).

   (e) The language containing all strings that do not contain 111 as a substring.

4. **Regular expressions II:** For each of the following languages over the alphabet $\{0, 1\}$, give a regular expression that describes that language, and briefly argue why your expression is correct.

   (a) All strings that do *not* contain 000 as a subsequence.

   (b) Strings in which every occurrence of the substring 00 appears before every occurrence of the substring 11.

   (c) Strings that do not contain the subsequence 010.

   (d) Strings that do not contain the subsequence 0101010.

   (e) Strings that do not contain the subsequence 10.