Assignments are due on the due date before class and have to include this cover page. Plagiarism in assignment answers will not be tolerated. By submitting their answers to this assignment, the authors named above declare that its content is their original work and that they did not use any sources for its preparation other than the class notes, the textbook, and ones explicitly acknowledged in the answers. Any suspected act of plagiarism will be reported to the Faculty’s Academic Integrity Officer and possibly to the Senate Discipline Committee. The penalty for academic dishonesty may range from failing the course to expulsion from the university, in accordance with Dalhousie University’s regulations regarding academic integrity.
**Question 1 (10 marks)** This question asks you to explore the benefits of splitting compilation into front end and back end.

(a) Draw a table listing for each of the six steps of compilation whether it is language-dependent or language-independent and whether it is platform-dependent or platform-independent, where “platform” refers to the combination of hardware and operating system the compiled code is to run on.

(b) Referring to the table you produced in (a), how would you characterize the work the front end/back end does?

(c) Now consider the task of constructing compilers for several different languages (say C, C++, Java, Ada, Eiffel, and Fortran) for several different platforms (say Windows, Linux, OS X, and Solaris). Explain how the split into front end and back end makes it easier to develop these compilers, compared to a monolithic compiler design.

**Question 2 (30 marks)** Do the following for each of the languages below:

- State whether or not it is regular.
- If you claim the language is regular,
  - Show how to construct the language from elementary regular languages (the empty language, the language containing the empty string, and the language containing a single one-character string) using the operations for combining regular languages into a new regular language (union, concatenation, and Kleene star).
  - Give a regular expression that matches exactly the strings in the language.
  - Provide a graphical representation of a DFA that recognizes the language.
- If you claim the language is not regular, give an intuitive argument why this is true.

The languages you are asked to consider are the following:

(a) All binary strings that represent numbers divisible by 4.

(b) All binary strings that do not contain any group of three consecutive 1s.

(c) All strings over the alphabet {a, b, c} that start with a group of as, followed by an arbitrary combination of bs and cs, and which contain as many as as bs and cs combined.