EECS 839
Spring 2020

EECS 839 Programming Project

due: April 28, 2020

Implement the MLEM2 algorithm for rule induction, based on concept approximations. **Your project should be implemented on one of the departmental computers, e.g., cycle 1, under Linux**. You may use any programming language you wish. Your program should accommodate symbolic and numerical attributes and attributes with missing attribute values.

Input data files will be in the LERS format. Input data files will start from the list of variable declarations. This list starts form "<", then a space, then a sequence of the following symbols: "a", "x", or "d", separated by spaces; then another space, the last symbol of the list is ">". You may ignore this line. The second list of the input data file starts from "["; then a space, then comes a list of attribute and decision names, separated by spaces; then another space, and then "]". The decision will be always the last variable, all remaining variables are attributes. The following part contains values of the attributes and decisions. Value "?" means the *lost* attribute value, "*" means the "*do not care*" condition and "–" means the *attribute-concept* value. The input data file may contain comments. A comment starts from "!"; everything that follows "!", until the end of the line, is the comment. Obviously, comments should be ignored during reading of data. Attribute values are separated by spaces. Spaces should be understood not only as ordinary spaces but also as white space characters, such as the end of line, tab, etc. Any line of the input data file may start from one or more spaces and one or more spaces may end it. Note that a "line" does not need to be a physical line (it is rather a paragraph). Examples of symbolic values are: *medium*, 12..14, 1.25..2.37, etc. Examples of numerical values are 42, –12.45, etc. Use *all cutpoints* strategy for numerical attributes.

First your program should ask the user for the name of the input data file. The expected response of the user is the name followed by pressing the <RETURN> key. The following question of the program is about the choice between lower or upper concept approximations. Then the program should ask for a name of the output data file (with rules).

Your program should create the output data file with rules, created by MLEM2, one rule per line, e.g.

```
2, 2, 2
(Ink-color, blue) & (Body-color, blue) -> (Attitude, plus)
1, 1, 1
(Ink-color, red) -> (Attitude, plus)
1, 3, 3
(Body-color, black) -> (Attitude, minus)
1, 3, 3
(Ink-color, black) -> (Attitude, minus)
```

General Remarks. Your program should be able to deal with unexpected answers of the user and not crush but rather repeat the question. Use of recursion is not encouraged. You may assume that the input data file does not contain errors. You should expect input data files of any size, with thousands examples and hundreds attributes.

Include all comments, including instructions about compiling and linking in a single file called **read.me**. Do not forget to include your name and KUID#. When you are ready to submit the project, send ALL necessary source files, makefile (if any), and the read.me file by e-mail. **Do not send object files, executable files, and test data files to the TA.** Time stamp of the e-mail will be used to decide late penalty, if applicable. Late projects will be accepted with 10% penalty per day up to five days.