EECS 22L: Chess Final Release

Prepared by: Mina Moghadam, Mihnea Chirila, Hamid Nejatollahi, Bo Tsai, Prof. Quoc-Viet Dang and Prof. Rainer Dömer

January 31, 2018

The final deliverable in the chess project is the release version of the chess program for the user, as well as the corresponding release version of the program source code and documentation. In contrast to the previous alpha version, this release version should be complete and fully functional. In particular, we expect the final version to follow the official rules of chess, so that a fair chess tournament can be played.

Note: Before submitting the packages, untar them at another location and cross check their compilation and execution. The extracted folder should strictly follow the hierarchy given below. Additionally for compiling and running the code, there should be no need for the user/developer to copy files from one folder to another (Paths in Makefile should accommodate the given hierarchy).

The credit will be awarded in the following way: 30% of the credit will be given for the packaging (including file hierarchy), the rest will be given on the content of each file and the performance of the program. For full credit (100%), your package should have the following minimum file hierarchy:

User/Customer Package (Chess_V1.0.tar.gz)

- **README** // author, version, date, general instructions · · ·

- **COPYRIGHT** // author and copyright

- **INSTALL** // installation instructions

- **bin/** // binary directory

    - *chess* : the executable chess program

- **doc/** // documentation directory

    - *Chess_UserManual.pdf*

Source Code Package (Chess_V1.0_src.tar.gz)

- **README** // author, version, date, general instructions, · · ·

- **COPYRIGHT** // author and copyright

- **INSTALL** // installation instructions, just let the user type `make` is enough!

- **Makefile** // top-level, tool-specific Makefile, should support: `make`, `make test`, `make clean`, `make tar` (tar the source code package); debug version is recommended but not mandatory

- **bin/** // binary directory, when the user types `make`, the executable or its symbolic link should be generated here, as well as logs

- **doc/** // documentation directory, report, etc.

    - *Chess_UserManual.pdf*

- – *Chess_SoftwareSpec.pdf*

- **src/** // source directory, with all source code and test files

  - – *chess.c*
  - – *rules.c* // maybe other modules and header files: AI.c, Piece.c, chess.h, rules.h
  - – *test_rulecheck.c* // check the output of the rules checker for a given input
  - – *test_boarddisplay.c* // check the display of a modified board

BONUS POINTS: Bonus points will be awarded in the following categories:

- **Tournament score:** The points gathered in the tournament by each team will be directly translated to bonus points.

- **Extra features:** Each working extra will receive bonus points.

- **Catch-up on previous assignments:** Files delivered in the previous assignments that have been updated according to the feedback received will be awarded extra points.

Notes:
README example: This is the final version of the chess software, to install, please type "make", for more detailed instructions, refer to the user manual.
INSTALL example: Type `tar -xvzf Chess_V1.0_src.tar.gz`, then `cd Chess_V1.0_src`, then `make`

The above is fine. Keep **README, INSTALL, COPYRIGHT** short. Guide the user to refer to the user manual or the developers to software spec for more details.

You can add other folders to your package if necessary, such as GUI libraries, resources (images/fonts/audio), etc.