

## Programming Language Course

13 oct. 2020

Please deliver on [\\srv-profs\Partage\\_Profs\body.e\Depots](\\srv-profs\Partage_Profs\body.e\Depots), in a folder named after your last name : (-3)

- A folder named **solitaire**, containing C code files and C header files with lowercase names that make sense (example : **main.c**, **cards.c**, **cards.h**, **user\_interface.c**, **user\_interface.h**, **game.c**, **game.h**)
- The code should be organized in the file in a logical way : for instance, no ncurses functions should be called outside the files where the user interface is managed (-1)
- The program should be compilable with the command **gcc -std=c89 -pedantic \*.c -lncurses -lsolitaire**.

Expected features : (-3)

- Explain the rules and/or provide a link to a complete explanation of the rules of the game
- Make the controls explicit in the interface
- Play an entire game using keyboard control
- Win or loose the game
- Play again

Bonus features : (+1)

- Three variants / difficulty levels
- Have a nice menu (not throwing directly the player into the game)
- Accept mouse inputs
- Save the game / Load the game (+2)
- Undo last move (just the last, not the previous ones) (+2)
- Display seed of current game and allow the user to choose the seed of its game (+2)

Coding safety : (-2)

- Free every memory that needs to be freed
- Check the success of every dynamic memory allocation
- Don't let the possibility of dereferencing an invalid pointer happen
- Don't let the possibility of accessing an array out of bounds happen
- Close every file you open
- Don't let type overflow happen

Coding style : (-2)

- Don't duplicate code
- Use function and variable names that make sense
- Use snake\_case
- Use comments to indicate anything that isn't obvious
- Use proper and consistent indentation
- Use spaces around operators
- Don't use goto or continue or break (or return inside a loop)