PA2 will be the same function as PA1: a program that plays a stick-picking game, *with the following significant changes:*

- the program must be written in C
- the user always goes first
- the user may be badly behaved: they might enter anything at anytime, so your program must be prepared to handle any input
- to determine how many sticks to start with, the game accepts either a command line parameter or program input. If a command line argument is given, make sure:
    - there is *exactly one* such argument
    - the argument is a valid integer
- The executable program must be named stick
- The initial number of sticks must be at least 10

If no command line arguments are present, read a number from the user, and make sure it is a valid, positive integer.

If any errors are detected in choosing the initial number of sticks, print a message and exit.

- when the user is specifying how many sticks they want to pick, make sure they enter a valid number. This means:
    - they don't just hit enter
    - they enter an integer
    - the integer is between 1 and 3

If there is any problem with their entry, ask them to re-enter, and **keep trying** until they make a valid entry.

Note that you **must** use sscanf to test all entries. Do not simply compare the character to '1' '2 and '3' or use "atoi"

The user might enter " 3" or "001" If sscanf accepts these as integers, then your program must also.

I **strongly** suggest modularizing your program. The ODPs will help with modularization, and well discuss the concept in class. Please ask questions if anything is not clear.

push to GITLab as PA2 (and grant me Reporter access) by the deadline (which is **8 am** on the due date).