CSIS 6033 – Lab 4

Symbolic Execution with KLEE

In this lab, you will be writing a basic program to allow a user to play the hangman game with the correct answer being = "WARHAWKS". In part 1 you will write a simple program that allows a user to guess the answer as a game. In part 2 you will have KLEE use symbolic execution to find a sequence of guesses that ends in a victory for the user.

## Part 1

Your program should allow a user to enter a single letter, one at a time, and then check if that letter exists in the answer char array "WARHAWKS". If the letter does not exist in the answer array, or if the user has already guessed that letter, the user gets a strike and a message should print indicating that their guess was wrong. After three strikes, the program should end and print out an appropriate "Game Over!" message. In the case that the user guesses an undiscovered letter in the answer char array, you should print a message indicating they guessed correctly. Once the user successfully solves the hangman puzzle, you should output "Congratulations! You Win!" and exit the program.

After the resolution of each user guess, you should print out the current state of the game using "?" in place of the letters that have not yet been guessed. For example "? - ? - ? - ? - ? - ? - ? - ?" could be used to represent the original puzzle. If the user successfully guesses W, then you should print the current state of the puzzle: "W - ? - ? - ? - ? - W - ? - ?"

You should demonstrate that your program works as described above by running it on your system and taking screenshots of a successful guess, incorrect guess, losing after three strikes, and winning the game.

## Part 2

***NOTE THAT YOU MIGHT RUN INTO TROUBLE WITH C++ SUPPORT WHEN RUNNING KLEE. IF YOU ARE HAVING ISSUES, I WOULD RECOMMEND YOU TO STICK TO C SUPPORTED METHODS/LIBRARIES, e.g. PRINTF() INSTEAD OF COUT.***

You should instrument your code to make the variable that represents the user's guess a symbolic value. You should also instrument your code to generate an assertion error, through klee, upon successfully winning the game. Compile your code using clang to obtain the bitcode and then run that bitcode through KLEE to have it solve a sequence of guesses that would lead to winning the game. When running KLEE, you should have it stop solving once the game winning assertion error is thrown. The following two links should prove useful:

- Using KLEE to solve a maze: https://feliam.wordpress.com/2010/10/07/the-symbolic-maze/
- KLEE command-line options: https://klee.github.io/docs/options/#:~:text=The%20general%20form%20of%20a,%5Bprogram%2Doptions%5D%20).

Once you have had KLEE generate a winning sequence of guesses you should take a screenshot of the number of instructions and paths, as well as the contents of the winning test case. You should also take screenshots of the commands you used to generate your LastNameHangman.bc file and to run klee.

Graduate students should also answer the following:

- Explain the meaning for each value associated with each "guess" made in the .ktest file for the winning solution, e.g. name, size, data, …, text. What does each value represent specifically?
- Explain how you would calculate the total number of possible sequences of guesses by a user for this problem? (You don't actually have to solve for a number)

Submit a zip file to Blackboard which contains your ~~LastNameHangman.cpp~~, the .ktest file of your winning sequence solved by KLEE, and a word document which includes the ~~screenshots of the conditions described above~~, including those demonstrating the normal operation of your simple program and ~~screenshots showing that you used KLEE to find the solution.~~

For full credit your code should compile, run as described, and be appropriately commented. If I need to know anything in particular about how I should compile your code, include that in your document. Make sure that your code comments include at the top: name, date, semester, course, description of file.

You should submit your solutions to Blackboard by **Thursday, July 30 at 11:59pm**. Your assignment should be of the format CSIS6033_Lab3_pape.zip, where you should substitute your own last name. Be sure to clearly indicate any referenced material that you used to complete this assignment.