CS4328: Project #2

Due on April, 26, 2019 at 11:59PM

Jamal Rasool

In Collaboration with:

Kristof York

Report.

1. Summary

    a. Within this project we explored how p-threads with task parallelism works. The way the program was structured was a top down approach in which required first the initialization of p-thread variables that would handle the locks and synchronization of the multiple threads. From there the code required us to generate a random deck based upon the provided value at run-time, then based upon that data generate a card deck, and assign the cards to each of the players. Within the game three rounds are played and the winner is picked out of it based on the cards. Once the winner has been picked then the output functions handle the output data with the provided information structure noted in the project. The main function of the program acts as the driver for all of the code and logic implementation.

2. Readme

    a. In order to run the code, the following commands in the image (figure 1) below will need to be run in order to compile the p-thread code. Note, the code should be able to work on C++11 without any issues, and the make file is already pre-configured to handle the program. Once the gcc command has compiled the code, then you must run the command in figure 2 in order to run the code. The value 1 in the image is the seed variable, you may change it as intended in order to get different outcomes of the results. In the case that there is no int value provided the program will automatically resort to a base value of 0 as a fallback. Once the program is completed running it will append the results to the report.txt file.

    Figure 1:

    Figure 2:

```
./a.out 1 # The one can be any value due to it being the random seed value
```

Data.

1. Please refer to the report.txt
   a. Within the report.txt file has the 5 different test runs with the following seed values
      i. 1
      ii. 51
      iii. 85
      iv. 921
      v. 4211
   b. Your able to try different seed values or even create your own script to run the data programmatically.