CS F372 Operating Systems
Second Semester 2021-22

Assignment 1

Important guidelines:

- The assignment is to be done in groups of 8
- All members of the group must contribute to the assignment. A demo-cum-viva session will be scheduled during which it will be ascertained that all members have contributed. If any member does not contribute, the member will be awarded 0 in the assignment
- All submissions will be passed through a code-similarity checker. If the codes of two or more groups match then all the group members will be summarily awarded 0 in the assignment. This is irrespective of if only one member of a group is the offender. There is no partial penalty for dishonesty. Lifting code from the Internet also constitutes cheating
- Honest but incorrect submissions will be awarded partial credit through the demo-cum-viva session
- Discussion is encouraged between groups, copying is strictly prohibited. Use Piazza for discussion. What is the difference? Check the last part of this page: https://www.cse.iitd.ac.in/~mausam/courses/col772/spring2019/

Problem Statement:

- Write a C program to solve an NxN sudoku puzzle by brute force. The program should be compatible on an Ubuntu machine running 18.04 or 20.04
    - 0<N<=36, N is a perfect square. The value of N will be supplied as a command line argument.
    - The grid to be solved will be supplied as a text file as a command line argument. Empty cells in the grid have 0. Numbers greater than 9 will be represented by the number itself. Example of a 9x9 grid: https://drive.google.com/file/d/1zfhPVD80gyxCbGYnvhFfrr3j7X1LFcqu/view?usp=sharing
    - Example of a 25x25 grid:
        1. https://drive.google.com/file/d/15rSuXJbEtJroHia0D1tNQVz8L-kkxN6O/view?usp=sharing
        2. https://drive.google.com/file/d/1592QBBbA1x_E5p8lxZk7hjtx_spfXyF3/view?usp=sharing
    - So, your program should run as : ./sudoku.out N input.txt
    - The program should print the solved grid on the standard output in a tab separated format, similar to the input.
    - A skeleton code has been provided here. Please keep the skeleton intact to ensure compatibility. Incompatible codes will be treated as incorrect. Skeleton: https://drive.google.com/file/d/1XR37jXw-oAqozUrmjTmpluXOP1yn7daT/view?usp=sharing

    - Please note that there is an extra tab before the newline at the end of each line. This is essential to include for compatibility
    - Your program should take advantage of parallelism offered by the Linux fork(), IPC and pthread libraries. It is compulsory to use either fork() or pthread or both. Your program should spawn multiple (as many as required) child processes and/or threads to divide and conquer the problem in parallel.

How and What to submit:

- Phase 1:
    - Weight: 10% (7% for implementation + 3% for demo and viva)
    - In this phase your code will be checked for correctness and that all the guidelines are followed
    - Submission will be through CMS
    - Deadline for Phase 1: 25th February 23:59 hours
- Phase 2:
    - Weight: 5%
    - Groups which submit a correct solution in Phase 1 are eligible
    - In this phase a leaderboard will be set up to rank the solutions by execution time, for a given fixed grid. The link to view the leaderboard will be shared. The leaderboard will be updated at a fixed interval everyday. The leaderboard can be viewed at http://responsible-tech.bits-hyderabad.ac.in/os202122s2/leaderboard.csv
    - Scoring will be based on the rank of a group on the leaderboard at the end of Phase 2
    - Submission will be through scp-ing the code to a server. Credentials will be shared with the groups.
    - Deadline: 28th February 22:00 hours
- File Naming Convention:
    - `groupX_assignment1.c` [Where X is the group no.]
- Late Day rules apply only to Phase 1, not Phase 2. Phase 2 starts and ends on the given dates.