Sabancı University

Faculty of Engineering and Natural Sciences

CS406/CS531 Parallel Computing / Parallel Processing and Algorithms

Spring 2022-2023

Homework #2

Due: 07/05/2023 - 23:59

PLEASE NOTE:

Your program should be a robust one such that you have to consider all relevant programmer mistakes and extreme cases; you are expected to take actions accordingly!

You HAVE TO write down the code on your own.
You CANNOT HELP any friend while coding.
Plagiarism will not be tolerated!!

1 Introduction

In this assignment, you will implement a parallel Kakuro Solver. Example Kakuro puzzles could be found at kakuros.com. The purpose of the homework is to implement a parallel algorithm for Kakuro Puzzle **as efficient as possible.** Note that the grading will be done based on the end-to-end execution times of implementations.

2 Kakuro Puzzle

Kakuro puzzles are similiar to crosswords, but you need to fill empty cells with digits(1-9) instead of letters. Each sequence of numbers, which summation is indicated by a corresponding hint, is called a **sum**. For every sum, each digit must be used at most once and their summation must be equal to their hint. A sum could be horizontal or vertical.

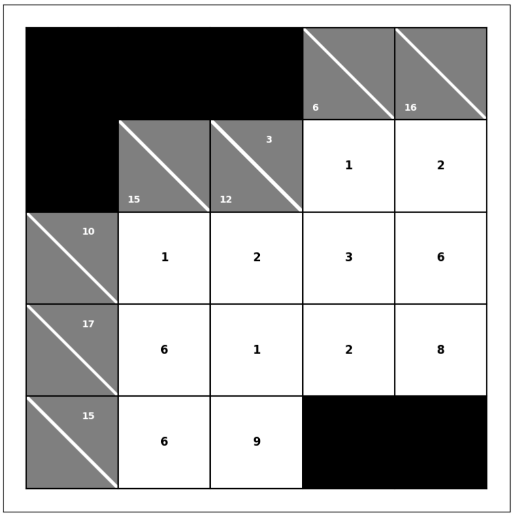

Figure 1: An example kakuro board

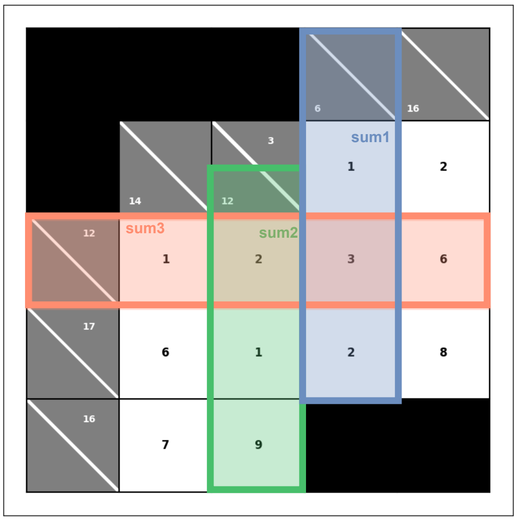

Figure 2: An example kakuro board with sums marked

3 Program Flow & Calculation

You will be reading puzzles from provided files with provided reader and calculate the solution of the given puzzle. You need to write a solution function that is marked with TODO in `kakuro_solver.cpp`. The filename of the matrix must be passed to your program by command line (use argc/argv).

For ease of use, a reader function that return `mat`, a simplifier that return `sol_mat` and vector of sums are provided to you within `kakuro_solver.cpp`. A `kakuro_explorer.py` is also provided to you to visually inspect your solutions. In any point of your execution, you can visualize your board by calling `sol_to_file` and giving resulted file to `kakuro_explorer.py`. (You can visualize your solutions on Nebula if you add -Y flag during ssh connection) You will also use this function to write your solutions to file. You can use `print_one_matrix()` function to keep of track of the status of your matrices during your execution. In the printed matrix, hints (gray cells) and black cells are represented with $-\mathbf{1}$'s and empty cells are represented with $-\mathbf{2}$'s.

You can use any other data type besides the provided ones for matrices and sums, custom constraint check functions, and any solution algorithm to achieve higher efficiency. (Hint: recursive backtracking algorithm may be useful).

With the homework, some sample boards are provided to you. Among these boards `board4_2` and `board5_2` are **extremely hard** and impossible to solve without smart parallel algorithms. It's understandable if you've only been able to solve other puzzles but try the best you can.

Your report will have equal importance as the implementation. A report of at least two pages should be submitted along with the code, including:

- A general explanation of the implementation and used algorithm, any heuristics etc.
- Execution times w.r.to sample boards and speed-up values. Time only `solution()` function. You can use `omp_get_wtime()`
- Tricks done for efficiency (providing spatial/temporal locality, preprocessing etc.)
- How you compile and run your program.

4 Some Remarks

In the grading process, three things will be checked:

- Correctness of your implementation

- Efficiency of your implementation
- How well the report is written

Your code will be compiled with the **-O3** optimization flag, so you are expected to include the relevant results in your report.
Submit only your `.cpp` file and indicate how you compile in report. **This is important:** use an HPC environment while working (e.g. Nebula). Do not submit Visual Studio solutions or any executable.
You can use ChatGPT as you wish for assistance. You can share your prompts within your report if you think it produces useful results.

5 What and Where to Submit (PLEASE READ, IMPORTANT)

Please don't forget to submit your code and the report together. Your **REPORT** must be a **pdf** file (preferable prepared by LaTeX but MS Word converted pdf's are also OK). It must contain the description of the optimizations you implemented, i.e., it must explain how you improved the performance, what was the timings before and after. You must do this for both of the –O0 and –O3 optimization options. Please see above what else do you need to include in the report.

The grading process is not automatic. However, the students are expected to strictly follow the guidelines in order to have a smooth grading process. If you do not follow these guidelines, depending on the severity of the problem created during the grading process, 5 or more penalty points are to be deducted from the grade. The name of your source code file that contains your program must be **kakuro_solver_hw2.cpp**. Similarly the report must be named as **report_hw2.pdf**.

Put both of these files into a folder named

SUCourseUserName_YourLastname_YourName_HWnumber

Your SUCourse user name is actually your SUNet username that is used for checking sabanciuniv e- mails. Do NOT use any spaces, non-ASCII and Turkish characters in the file name. For example, if your SUCourse user name is cago, name is Çağlayan, and last name is Özbugsızkodyazaroğlu, then the folder name must be:

cago_Caglayan_Ozbugsizkodyazaroglu_hw1

Do not add any other character or phrase to the folder name. Make sure that it contains the last version of the source code and the report. Compress this folder using a zip program. Please use "zip" compression. **"rar" or another compression mechanism is NOT allowed. Please make sure that you include both of the files in the compressed folder.**

You will receive no credits if your compressed folder does not expand or it does not contain the correct files. The name of the zip file should be as follows:

SUCourseUserName_YourLastname_YourName_HWnumber.zip

For example zubzipler_Zipleroglu_Zubeyir_hw1.zip is a valid name, but

hw1_hoz_HasanOz.zip, HasanOzHoz.zip

are **NOT** valid names. **Submit via SUCourse ONLY!** You will receive no credits if you submit by other means (e-mail, paper, etc.).

Successful submission is one of the requirements of the homework. If, for some reason, you cannot successfully submit your homework and we cannot grade it, your grade will be 0.

Good Luck!

CS406-531 Team (Fatih Taşyaran, Kamer Kaya)