Operating Systems Programming Projects for Chap.4 & Chap. 6

Team members:

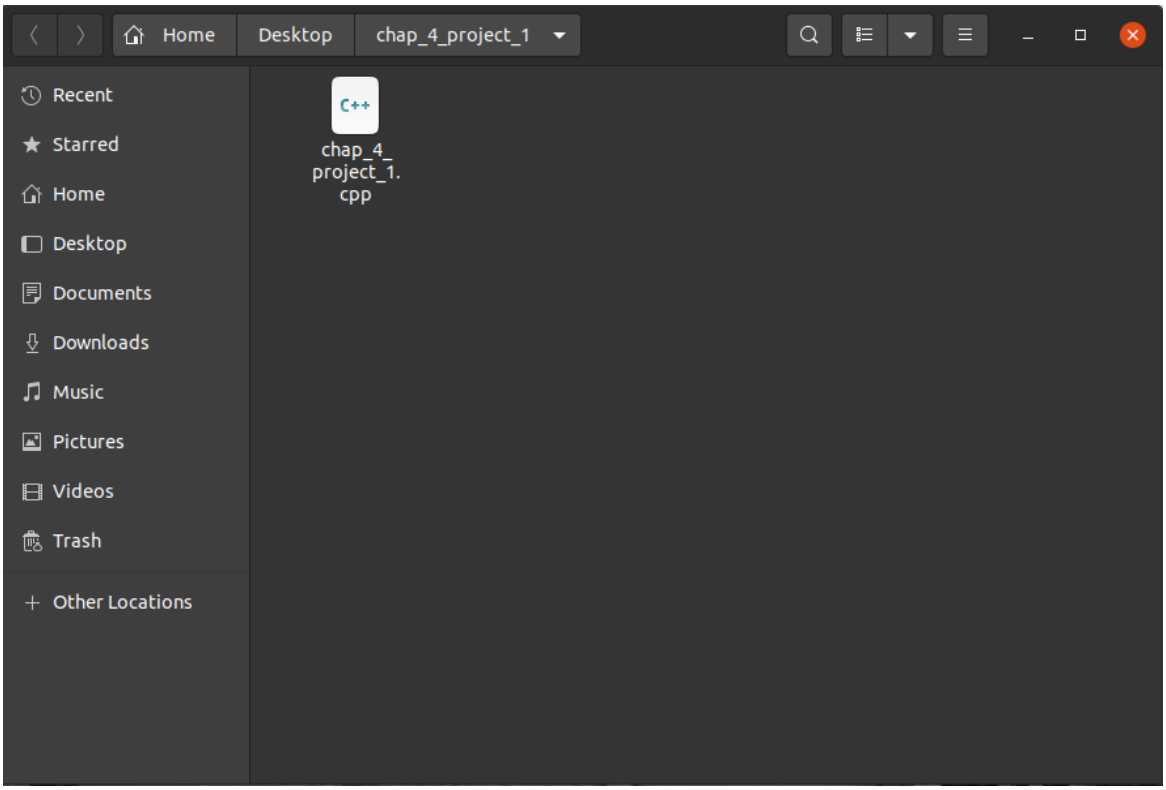
資工三 方文昊 **108590048** **(Wrote and designed the programs and documentation)**
資工三 鄭琳玲 **108590056** **(Wrote and designed the programs and documentation)**
資工三 黃聖耀 **108590061** **(Wrote and designed the programs and documentation)**
電資三 李以謙 **108820021** **(Wrote and designed the programs and documentation)**

Folder name description:

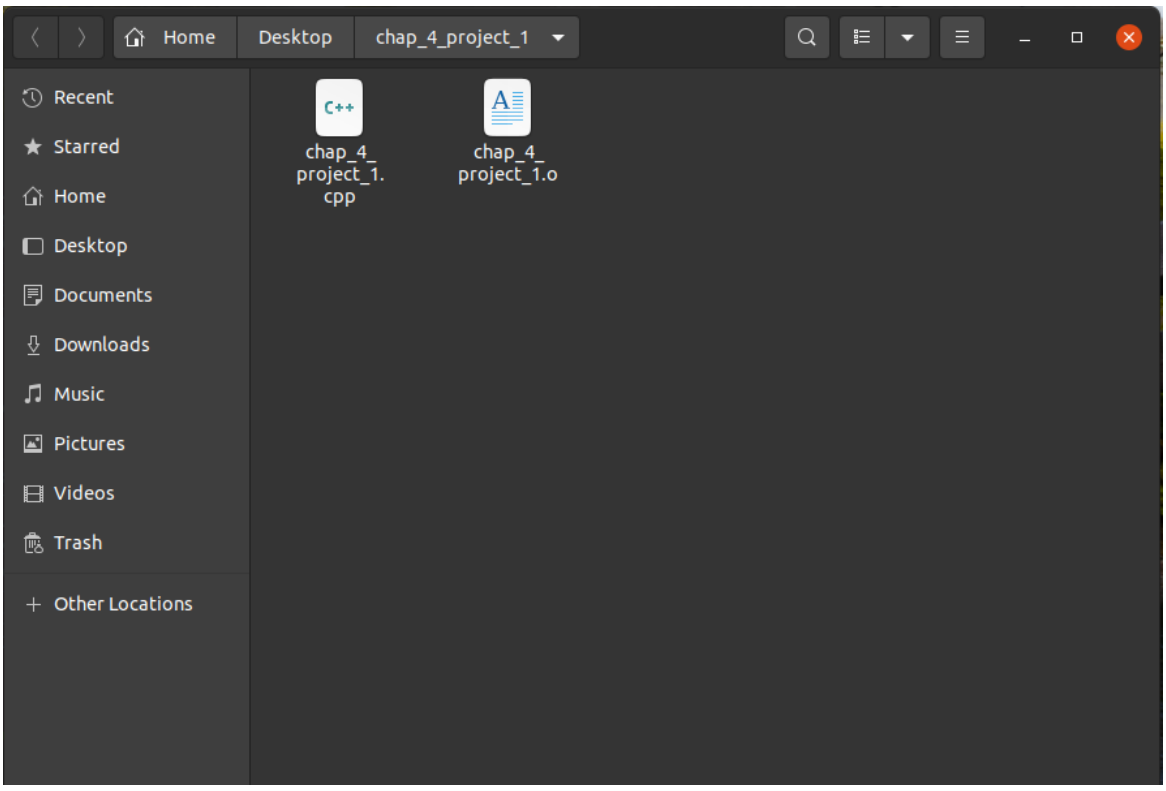
- **chap_4_project_1** → Programming Projects Chap. 4, Project 1
- **chap_6_project_1** → Programming Projects Chap. 6, Project 1

Programming Project for Chap. 4
Project 1

In Linux, go to the chap_4_project_1 folder, then open the linux terminal inside of the folder,



then in the linux terminal, input **g++ chap_4_project_1.cpp -o chap_4_project_1.o -pthread**



Then input **./chap_4_project_1.o**

```
star@ubuntu:~/Desktop/chap_4_project_1$ g++ chap_4_project_1.cpp -o chap_4_project_1.o -pthread
star@ubuntu:~/Desktop/chap_4_project_1$ ./chap_4_project_1.o
Below are the examples of valid and invalid sudoku that are
checked or validated by the sudoku validator.

Valid sudoku checking

{7, 2, 3, 5, 4, 9, 8, 6, 1}
{9, 5, 1, 8, 3, 6, 7, 4, 2}
{8, 6, 4, 7, 2, 1, 9, 5, 3}
{6, 9, 8, 4, 7, 3, 2, 1, 5}
{5, 4, 7, 9, 1, 2, 6, 3, 8}
{3, 1, 2, 6, 8, 5, 4, 9, 7}
{4, 8, 9, 1, 5, 7, 3, 2, 6}
{2, 7, 5, 3, 6, 4, 1, 8, 9}
{1, 3, 6, 2, 9, 8, 5, 7, 4}

Sudoku is valid.
```

It will also show the invalid sudoku conditions examples

```
Several invalid sudoku conditions:

Column checking

{7, 2, 3, 5, 4, 9, 8, 6, 1}
{9, 5, 1, 8, 3, 6, 7, 4, 2}
{8, 6, 4, 7, 2, 1, 9, 5, 3}
{6, 9, 8, 4, 7, 3, 2, 1, 5}
{5, 4, 7, 9, 1, 2, 6, 3, 8}
{3, 1, 2, 6, 8, 5, 4, 9, 7}
{4, 8, 9, 1, 5, 7, 3, 2, 6}
{2, 7, 8, 3, 6, 4, 1, 8, 9}
{1, 3, 6, 2, 9, 8, 5, 7, 4}

there are same numbers in the column which are [3][2] and [7][2], therefore: Sudoku is invalid.

Row checking

{7, 2, 3, 5, 4, 9, 8, 6, 1}
{9, 5, 1, 8, 3, 6, 7, 4, 2}
{8, 6, 4, 7, 2, 1, 9, 7, 3}
{6, 9, 8, 4, 7, 3, 2, 1, 5}
{5, 4, 7, 9, 1, 2, 6, 3, 8}
{3, 1, 2, 6, 8, 5, 4, 9, 7}
{4, 8, 9, 1, 5, 7, 3, 2, 6}
{2, 7, 5, 3, 6, 4, 1, 8, 9}
{1, 3, 6, 2, 9, 8, 5, 7, 4}

 there are same numbers in the row which are [2][3] and [2][7], therefore: Sudoku is invalid.

Grid checking

{7, 2, 3, 5, 4, 9, 8, 6, 1}
{9, 5, 1, 8, 3, 6, 7, 4, 2}
{8, 6, 4, 7, 2, 1, 9, 5, 3}
{6, 9, 8, 4, 7, 3, 2, 1, 5}
{5, 4, 7, 9, 1, 2, 6, 3, 8}
{3, 1, 2, 6, 8, 5, 4, 9, 7}
{4, 8, 9, 1, 5, 7, 3, 2, 6}
{2, 7, 5, 3, 6, 4, 1, 8, 9}
{1, 3, 7, 2, 9, 8, 5, 7, 4}

there are same numbers in the grid which are [7][1] and [8][2], therefore: Sudoku is invalid.
```

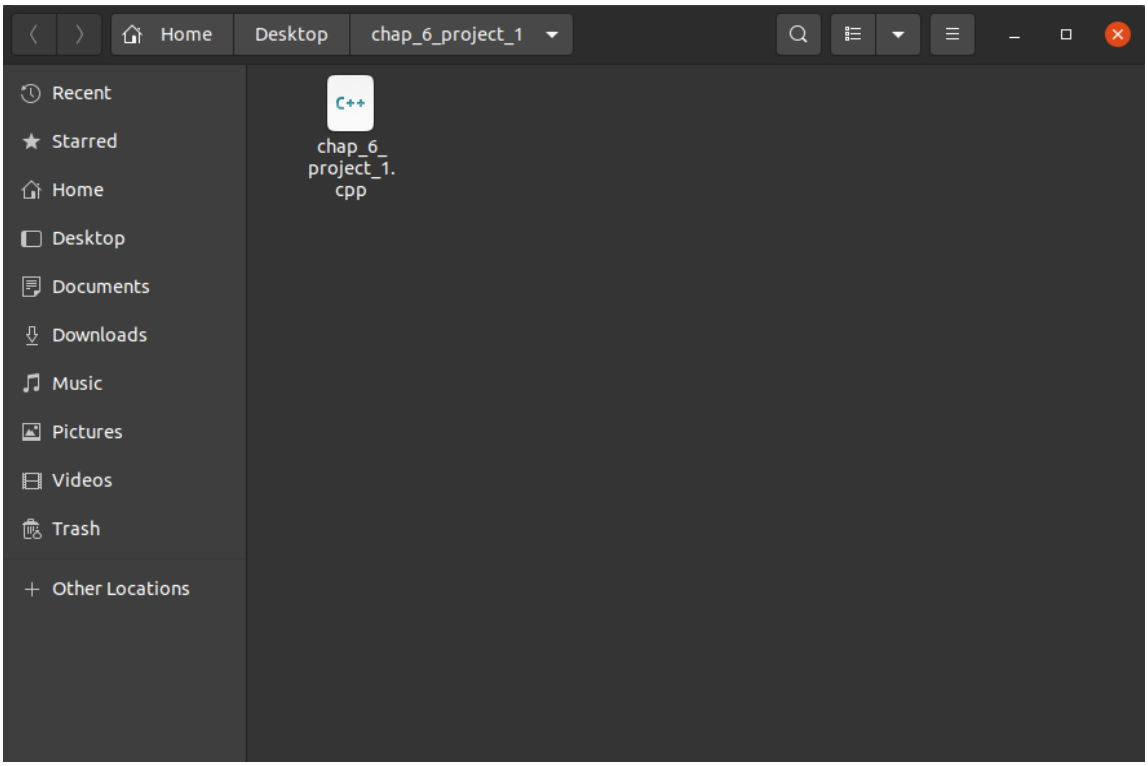
And some others invalid sudoku conditions

```
Some others invalid sudoku conditions checking:

Number less than one checking

{7, 2, 3, 5, 4, 9, 8, 6, 1}
{9, 5, 0, 8, 3, 6, 7, 4, 2}
{8, 6, 4, 7, 2, 1, 9, 5, 3}
{6, 9, 8, 4, 7, 3, 2, 1, 5}
{5, 4, 7, 9, 1, 2, 6, 3, 8}
{3, 1, 2, 6, 8, 5, 4, 9, 7}
{4, 8, 9, 1, 5, 7, 3, 2, 6}
{2, 7, 5, 3, 6, 4, 1, 8, 9}
{1, 3, 6, 2, 9, 8, 5, 7, 4}

there is a zero number in [1][2], therefore: Sudoku is invalid.

Number less than one checking

{7, 2, 3, 5, 4, 9, 8, 6, 1}
{9, 5, 1, 8, 3, 6, 7, 4, 2}
{8, 6, 4, 7, 2, 1, 9, 5, 3}
{6, 9, 8, 4, 7, 3, -2, 1, 5}
{5, 4, 7, 9, 1, 2, 6, 3, 8}
{3, 1, 2, 6, 8, 5, 4, 9, 7}
{4, 8, 9, 1, 5, 7, 3, 2, 6}
{2, 7, 5, 3, 6, 4, 1, 8, 9}
{1, 3, 6, 2, 9, 8, 5, 7, 4}

there is a negative number in [3][6], therefore: Sudoku is invalid.

Number more than nine checking

{7, 2, 3, 5, 4, 9, 8, 6, 1}
{9, 5, 1, 8, 3, 6, 7, 4, 15}
{8, 6, 4, 7, 2, 1, 9, 5, 3}
{6, 9, 8, 4, 7, 3, 2, 1, 5}
{5, 4, 7, 9, 1, 2, 6, 3, 8}
{3, 1, 2, 6, 8, 5, 4, 9, 7}
{4, 8, 9, 1, 5, 7, 3, 2, 6}
{2, 7, 5, 3, 6, 4, 1, 8, 9}
{1, 3, 6, 2, 9, 8, 5, 7, 4}

there is a number which is more than nine in [1][8]: Sudoku is invalid.
```

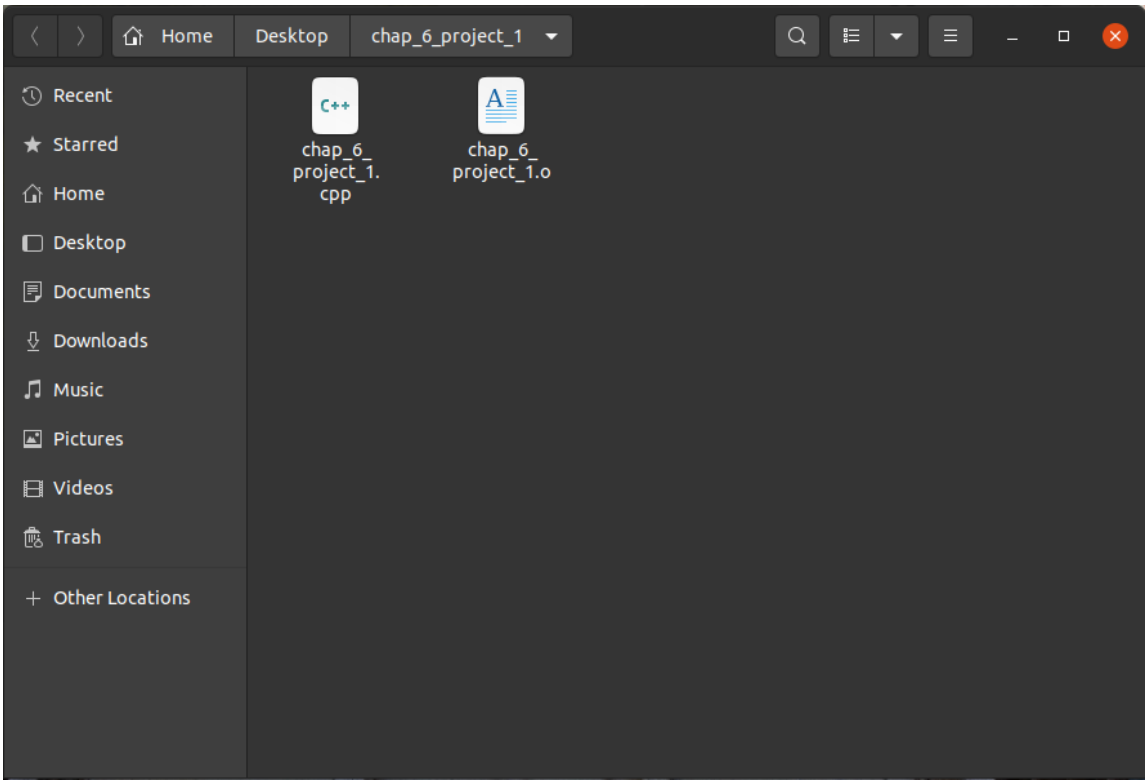
The multithread is use to verify or validate whether the solution to the Sudoku puzzle is valid or invalid by checking the sudoku row, column , grid. And this sudoku validator also include some tothers invalid sudoku checking such as if there exist number less than one such as zero or negative numbers and also number which is more than nine.

Programming Project for Chap. 6
Project 1

In Linux, go to the chap_6_project_1 folder, then open the linux terminal inside of the folder,



then in the linux terminal, input **g++ chap_6_project_1.cpp -o chap_6_project_1.o -pthread**



Then input **./chap_6_project_1.o 3,** where 3 is the number of the students, as shown in the screenshot example below**.** And press **Ctrl + c** to stop the running program.

```
star@ubuntu:~/Desktop/chap_6_project_1$ g++ chap_6_project_1.cpp -o chap_6_project_1.o -pthread
star@ubuntu:~/Desktop/chap_6_project_1$ ./chap_6_project_1.o 3
There are 3 students.
Student 0 is doing the programming assignment.
Student 1 is doing the programming assignment.
Student 2 is doing the programming assignment.
Student 0 needs help from the TA.
The student is sitting on the chair. The remaining chairs: 2
A student awakens the TA.
The student left the chair. The remaining chairs: 3
The TA is currently helping the student.
Student 0 is getting help from the TA.
Student 1 needs help from the TA.
The student is sitting on the chair. The remaining chairs: 2
Student 2 needs help from the TA.
Student 2 is sitting on a chair waiting for the TA to finish.
The student is sitting on the chair. The remaining chairs: 1
Student 0 left TA room.
Student 0 is doing the programming assignment.
The student left the chair. The remaining chairs: 2
The TA is currently helping the student.
Student 1 is getting help from the TA.
Student 0 needs help from the TA.
Student 0 is sitting on a chair waiting for the TA to finish.
The student is sitting on the chair. The remaining chairs: 1
Student 1 left TA room.
Student 1 is doing the programming assignment.
The student left the chair. The remaining chairs: 2
The TA is currently helping the student.
Student 2 is getting help from the TA.
```

In this programming projects for Chapter 6, project 1, the sleeping teaching assistant, there is a room with 1 desk with a chair and computer and hallway with 3 chairs.