CS6456 (F2015) Operating Systems

Project 3: Check Valid Sudoku Solution With Multithread

Chenxi Guo

1. Function

(1) int parseInput(char* input, char(*output)[9])
turn input string into a 2-dimensional array.

(2) void *checkSubgrid(void *inputArg)

a) check each subgrid and print all the missing number in this subgrid.
b) If there is any number is missed is this subgrid, set the flag to -1, which means this solution is already invalid.

(3) void *checkRow(void *inputArg)

a) check each row and print, if a number is missed in a row, print the number and the row.
b) As required, if several numbers are missed in a row, the program should print several times.
c) If there is any number is missed in any row, set the flag to -1, which means this solution is already invalid.

(4) void *checkColumn(void *inputArg)

a) check each column and print, if a number is missed in a column, print the number and the row.
b) As required, if several numbers are missed in a column, the program should print several times.
c) If there is any number is missed in any column, set the flag to -1, which means this solution is already invalid.

2. threads arrangement

thread tid[0] is used to check each row; thread[1] is used to check each column; thread[2] to thread[11] are used to check 9 subgrid.

3. data structure

Since pthread_create() can only pass one parameter to a function, I define a type - argument as the parameter of the function.

```
typedef struct{
        int* flag_p;
        char (*wgrid)[9];
        parameters para;
}argument;
```

In this struct flag_p is used to record whether the solution is still valid during the check process. Wgrid is used to pass the input solution to functions, para is another struct type-partameters, which is used to pass the starting solution of the subgrid to the function. The type parameters looks like this

```
typedef struct{
        int row;
        int column;
     }parameters;
```

4. the structure of the program

In the main function, First I use fgets() to get the input of solution file. I suppose the input is 1 2 3 4 5 6 7 8 9 2 3 4 5 6 7 8 9 1 3 4 5 6 7 8 9 1 2 4 5 6 7 8 9 1 2 3 5 6 7 8 9 1 2 3 4 6 7 8 9 1 2 3 4 5 7 8 9 1 2 3 4 5 6 8 9 1 2 3 4 5 6 7 9 1 2 3 4 5 6 7 8, eight-one number with ‘ ’ as separation. (if there is not standard input during the test , you can email me to make the change for it.)

Secondly, function parseInput is called to store the input is a 2-dimensional array.

Then, the program calls the pthread_create to run the function checkrow and pass the data to the function.

After that, the program calls the pthread_create to run the function checkColumn and pass the data to the function.

The program then call enter the loop of calling pthread_create and call function checkSubgrid for 9 times.

Finally, the process waits for each thread finishing its work.