

COMP 281 Assignment1

Yifan Xu

Student ID : 201377026

7th, March, 2019

Contents

Problem 1081-----	1
Problem 1084-----	2

Problem 1081 Game of life

Program simulates the game called “Game of life”. Firstly, the programs take four inputs: the number of rows, the number of columns and the number of generations, followed by the original cell tables. Its constants alive cell as 'X', dead cell as '.'. The output of the program is the cell state table after several generations. There one main() function and five sub-functions: ask_table(), print_Table(), getValue(), count_Value(), neighbors().

In the `ask_table()` function, it takes table rows, table columns, and pointers to pointers of table. Using two for loop aims to initialize the table of cells. In `print_Table()` function, it also takes table rows, table columns, and pointers to pointers of table. Loop through the table matrix, and print out the state of each cells in the table.

Applying `getValue()` function is to return the value of surrounding neighbors. It takes searching row and col, table rows, table columns and pointers to pointers of table. If statement aims to check whether the cell is alive or dead and whether the search row or column is out of boundary or not. If cell is alive, return value 1, otherwise return value 0. `Count_Value()` function also takes searching row and col, table rows, table columns and pointers to pointers of table. It counts the surrounding neighbor cells and return the number of cells that matches the rule. In `neighbors()` function, it takes table rows, table columns, table and pointers to pointers of new table aiming to deal with the state of cells according to the game rule and obtain the distribution of updated cell table. Loop through the table matrix and call the `count_Value()` function to deal the state of cells. After a generation, update the table state via two for loops.

In the main function, the program allows user input the number of table rows, number of table columns and number of generations, followed by two pointers to pointers of tables., one is original and another is new table for updating. Then it calls the `ask_table()` function aims to initialize table. Using for loop is to get the cells distribution after passing the specific generations. Finally, print out the result and free the memory of two tables.

Problem 1084 Highway Lite

Program simulates a highway and takes four inputs: the number of rows, the number of columns and the number of timesteps, followed by a table for cars entering the highway. Its constants car as '1', empty space as '.'. The output of the program is the highway state table after several generations. There one `main()` function and five sub-functions: `car_Entering()`, `highway_situation()`, `init_Table()`, `move_Table()`, `print_Table()`.

In the `init_Table()` function, it takes highway table rows, highway table columns, and highway table pointer. Using two for loop aims to initialize the table of highways. `Move_Table()` takes highway's number, highway table rows, highway table columns, and pointer to pointer of highway table. Loop through the highway table matrix and move the table to the right step by step. In `print_Table()` function, it highway table rows, highway table columns, and pointers to

pointers of highway table. Loop through the highway table matrix, and print out the state of highway at the timestep.

In `car_Entering()` function, it takes the current time of car's running, highway's number, and the table pointer of car entering the highway, followed by the number of cars. Applying for loop and two if statements is to loop all the cars situation, including checking car enters which highway and the entering time. If car enters, return value 1, otherwise return value 0. `Highway_situation()` function also takes highway table rows, highway table columns and pointers to pointers of highway table. Using while loop allows user input the table of car entering, including time and highway's number. Then it implements two for loops and if statements to obtain and update the table of car entering.

In the main function, the program allows user input the number of highway table rows, number of highway table columns and number of timesteps, followed by creating the pointers to pointers of highway table. Then it calls the `init_table()`, `highway_situation()`, `print_Table()` methods. Finally, print out the result and free the memory of highway table.