HW5 (Due 2018/04/09)

1. Write a program to search a target in an $n \times m$ matrix of integers, where $1 \leq n, m \leq 100$. The program prompts users to input n, m, the elements of an $n \times m$ matrix and targets to search. If a searching target exists in the matrix, the program displays the position of target; otherwise, displays "Not found".

   Sample I/O: (The italics for program output and boldfaces for user input)

Please input n and m: **3 4** *Please input the elements of matrix n*m by row major:* *2 3 4 5* *1 2 3 4* *5 6 7 8* *Please input a target to search (<ctrl>-d to exit):* **8** *At row 2 and column 3.* *Please input a target to search (<ctrl>-d to exit):* **16** Not found. *Please input a target to search:* **<ctrl>-d**

   Bonus:

   1. Write a function **int search(int data[][100], int n, int m, int target, int &t_row, int &t_col)** that returns 1 if **target** is found in **n×m** array **data,** 0 otherwise. In addition, when **target** is found, **t_row** and **t_col** refer to the row number and column number of the target in array **data**, respectively.

2. Write a program to sort elements of an **n×m** array with top-down&left-right style. Your program must use a function **void m_sort(int data[][100], int n, int m, int order)** to sort elements of **n×m** array **data** with top-down&left-right style. If **order** is 1, the function sorts the elements in the non-increasing order. If **order** is 2, the function sorts the elements in the non-decreasing order.

   Sample I/O: (The italics for program output and boldfaces for user input)

Please input n and m: **3 4** *Please input the elements of matrix n*m by row major:* **2 3 4 5** **6 1 10 14** **30 25 13 9** *Select sorted order: 1) non-increasing order, 2) non-decreasing order (<ctrl>-d to exit):* **2** *The sorted matrix in the non-decreasing order:* *1 2 3 4*

5 6 9 10
13 14 25 30

Bonus:

You program can display the sorted elements with either top-down&left-right or snake-like style. Your program must use a function **void ms_sort(int data[][100], int n, int m, int order)** to sort elements of **n×m** array **data** with snake-like style. If **order** is 1, the function sorts the elements in the non-increasing order. If **order** is 2, the function sorts the elements in the non-decreasing order.

Sample I/O: (The italics for program output and boldfaces for user input)

Please input n and m: **3 4**
*Please input the elements of matrix n*m by row major:*
2 3 4 5
6 1 10 14
30 25 13 9
Select sorted style: 1) top-down&left-right*, 2) snake-like (<ctrl>-d to exit):* **2**
Select sorted order: 1) non-increasing order, 2) non-decreasing order (<ctrl>-d to exit): **2**
The sorted matrix in the non-decreasing order:
1 2 3 4
10 9 6 5
13 14 25 30