2.1

(1)

float N[n][n];

for(int i=0;i<n;i++)

for(int j=0;j<n;j++)

cin>>N[j][i];

then Matrix N is the transpose matrix of M.

(2)

Ex: 1 0 0 0

2 3 0 0

4 5 6 0

7 8 9 10...

int N[n], k=0;

for(int i=0; i<m; i++)

for(int j=0; j<=i; j++)

N[k++]=A[i][j];

A[m][m] can be stored as N[n]={1,2,3,4,5,6,7,8...}

then I can get/put A[i][j] from N[(i)\*(i-1)/2+j]

and I just use space = the lower triangular matrix A .

2.2

(1) 我分別將四項類別的資料(users/items/results/Unix-timestamp)存進structure中，並使用vector來儲存每筆資料，我發現當我以Users為優先排列時(Users/Items/Unix-timestamp)，我能輕易且有效率地用lower\_bound寫出“accept”, ”items” , “ratio”，但“users” 及 “findtime\_items”便會遇上麻煩，所以我也將資料以Items為優先排列以處理這兩個函數。

(2) It took 1m22.407s for me convert the raw data to my data structure on the 217workstation linux10.csie.ntu.edu.tw.

(3) 在寫code時，我有自己製造一筆小資料，來測試是否我的函數正確，或是有哪些部份需要修正。等code大致完成後，才放上工作站上跑範測，此外也有好心的同學生測姿讓我能測試，而我也因此找出了bug。