Ministry of Education and Science of Ukraine

National Technical University of Ukraine

«Kyiv Polytechnic Institute. Igor Sikorsky »

Faculty of Informatics and Computer Technologies

Department of Computer Engineering

LAB № 2

from the discipline "Theory of Algorithms"

on the topic «Decomposition method. Search for inversions»

PERFORMED BY:

1st year student

group ІП-93

Dominskyi V.O.

The credit - 9311

Variant – 11

CHECKED:

Associate Professor of OT

c.t.s.,s.r.

Antoniuk А.І.

Kyiv - 2020

**TASK**

**Goal:** using the decomposition method to develop an algorithm and find the number of degrees of liking (number of inversions).

**Option task:** **Use a decomposition method** to develop an algorithm that will solve the following problem.

Incoming data. Matrix D of natural numbers of dimension um, where u are the number of users, m - number of films. Each element of the matrix D [i, j] indicates the position of the movie j in the list of user preferences i. Another input is the x number of the user, which will be compared with the numbers of all other users.

Output data. List of ascending second element of pairs (i, c), where i is the user number, c is a number indicating the degree of similarity of the users' preferences x and c (number of inversions).

**CODE**

'use strict'

**const** Matrix **=**

**[**

**[**3**,** 7**,** 1**,** 12**,** 1**],**

**[**9**,** 11**,** 2**,** 14**,** 3**],**

**[**7**,** 6**,** 5**,** 4**,** 1**],**

**[**12**,** 10**,** 11**,** 6**,** 3**],**

**];**

**const** Num **=** 3

console**.**log**(**NumberOfInversions**(**Num**,** Matrix**));**

**function** NumberOfInversions**(**X**,** Matrix**)**

**{**

**let** N **=** Matrix**[**X**].**length**;**

**let** Final **=** **[];**

**for** **(const** Line **in** Matrix**)** **{**

**if** **(**Line **!=** X**)**

**{**

**let** Column **=** 0**;**

**let** ColumnArray **=** Matrix**[**Line**].**slice**();**

**for** **(let** Y **=** 0**;** Y **<** N**;** Y**++)**

**{**

**for** **(let** Z **=** 0**;** Z **<** N**;** Z**++)**

**{**

**if** **(**Matrix**[**X**][**Y**]** **===** ColumnArray**[**Z**])**

**{**

**if** **(**Y **!=** Z**)**

**{**

**let** NotPermanent **=** ColumnArray**[**Y**];**

ColumnArray**[**Y**]** **=** ColumnArray**[**Z**];**

ColumnArray**[**Z**]** **=** NotPermanent**;**

Column**++**

**};**

**};**

**};**

**};**

Final.push([Line, Column]);

}

}

return Final.sort(Compare);

}

function Compare(a, b) {

if (a[1] === b[1])

{

return 0;

}

else

{

return (a[1] < b[1]) ? -1 : 1;

}

}

**RESULTS OF THE PROGRAM WORK**

Output Array: [[ '1', 1 ],[ '2', 1 ],[ '0', 2 ]]

**CONCLUSIONS**

I got acquainted with the topic of laboratory work.

Have acquired relevant work skills.

An appropriate test program has been developed.

The results of the successful operation of the test program above confirm the correctness of the chosen decisions, the ultimate goal of the work is achieved.