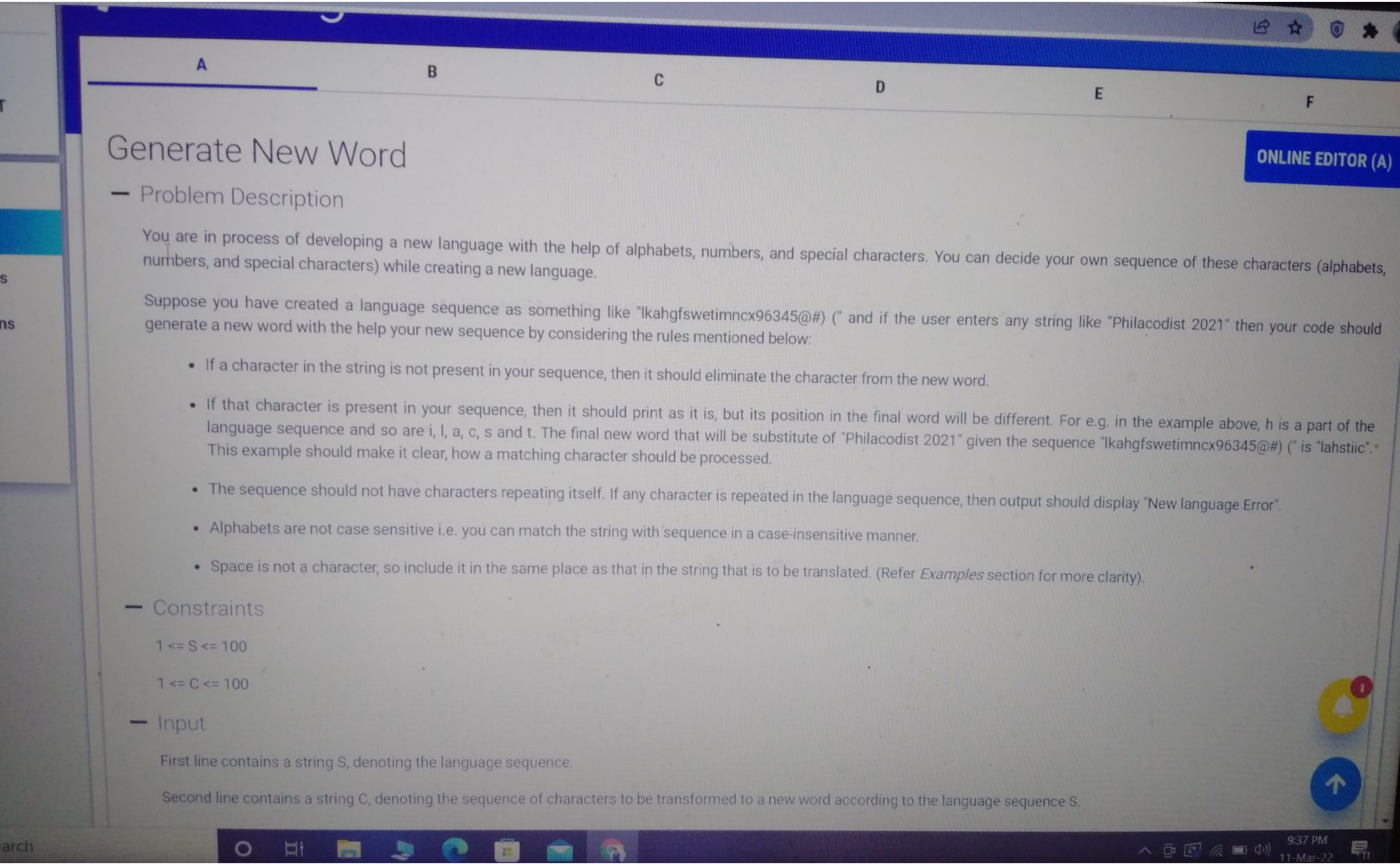


Generate New Word

— Problem Description

You are in process of developing a new language with the help of alphabets, numbers, and special characters. You can decide your own sequence of these characters (alphabets, numbers, and special characters) while creating a new language.

Suppose you have created a language sequence as something like "lkahgfswetimncx96345@#) (" and if the user enters any string like "Philacodist 2021" then your code should generate a new word with the help your new sequence by considering the rules mentioned below:

- If a character in the string is not present in your sequence, then it should eliminate the character from the new word.
- If that character is present in your sequence, then it should print as it is, but its position in the final word will be different. For e.g. in the example above, h is a part of the language sequence and so are i, l, a, c, s and t. The final new word that will be substitute of "Philacodist 2021" given the sequence "lkahgfswetimncx96345@#) (" is "lahstiic". This example should make it clear, how a matching character should be processed.
- The sequence should not have characters repeating itself. If any character is repeated in the language sequence, then output should display "New language Error".
- Alphabets are not case sensitive i.e. you can match the string with sequence in a case-insensitive manner.
- Space is not a character, so include it in the same place as that in the string that is to be translated. (Refer *Examples* section for more clarity).

— Constraints

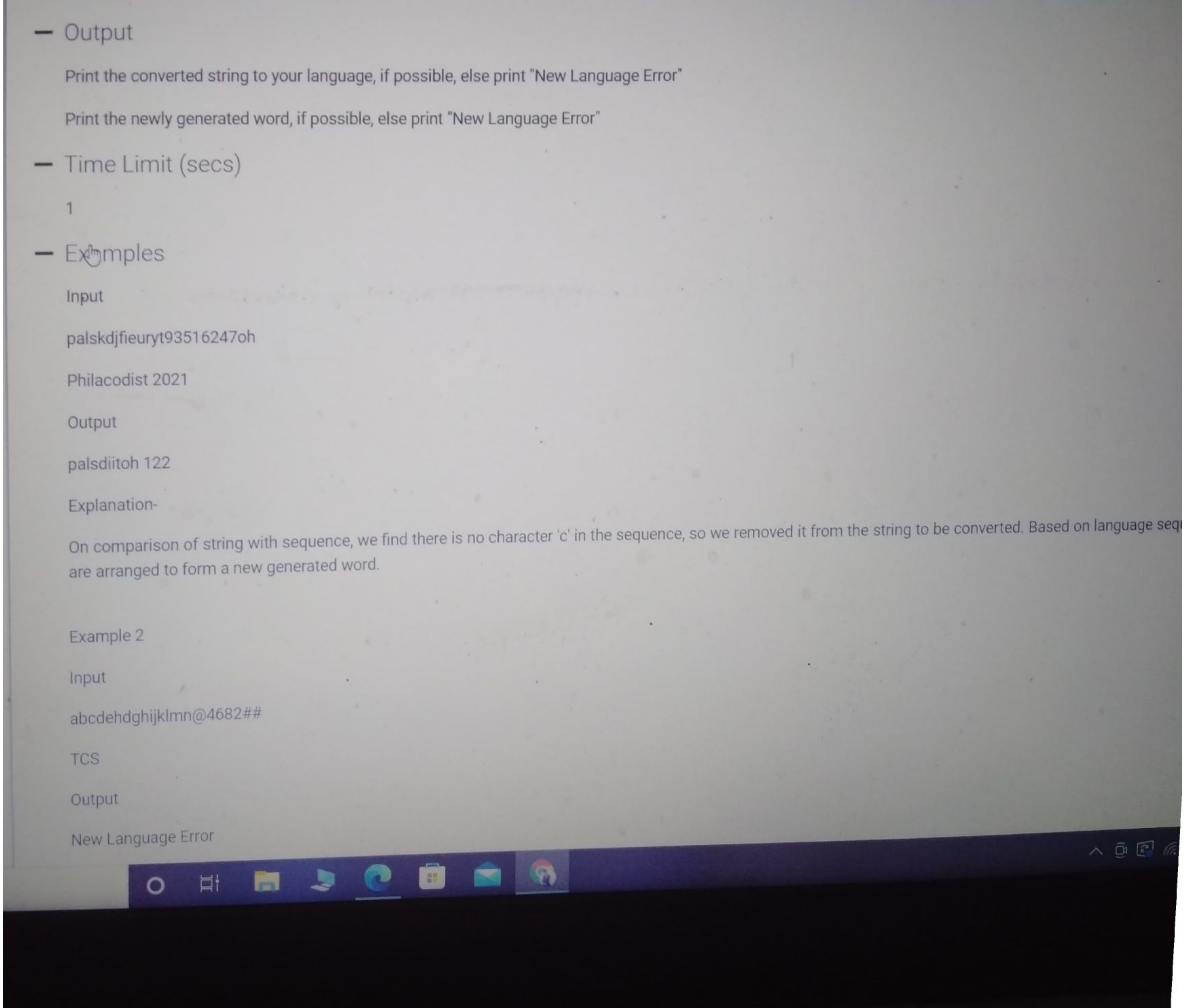
1 <= S <= 100

1 <= C <= 100

— Input

First line contains a string S, denoting the language sequence.

Second line contains a string C, denoting the sequence of characters to be transformed to a new word according to the language sequence S.



Output

Print the converted string to your language, if possible, else print "New Language Error"

Print the newly generated word, if possible, else print "New Language Error"

Time Limit (secs)

1

Examples

Input

```
palskdjfieuryt93516247oh
Philacodist 2021
```

Output

```
palsdiitoh 122
```

Explanation-

On comparison of string with sequence, we find there is no character 'c' in the sequence, so we removed it from the string to be converted. Based on language seq
are arranged to form a new generated word.

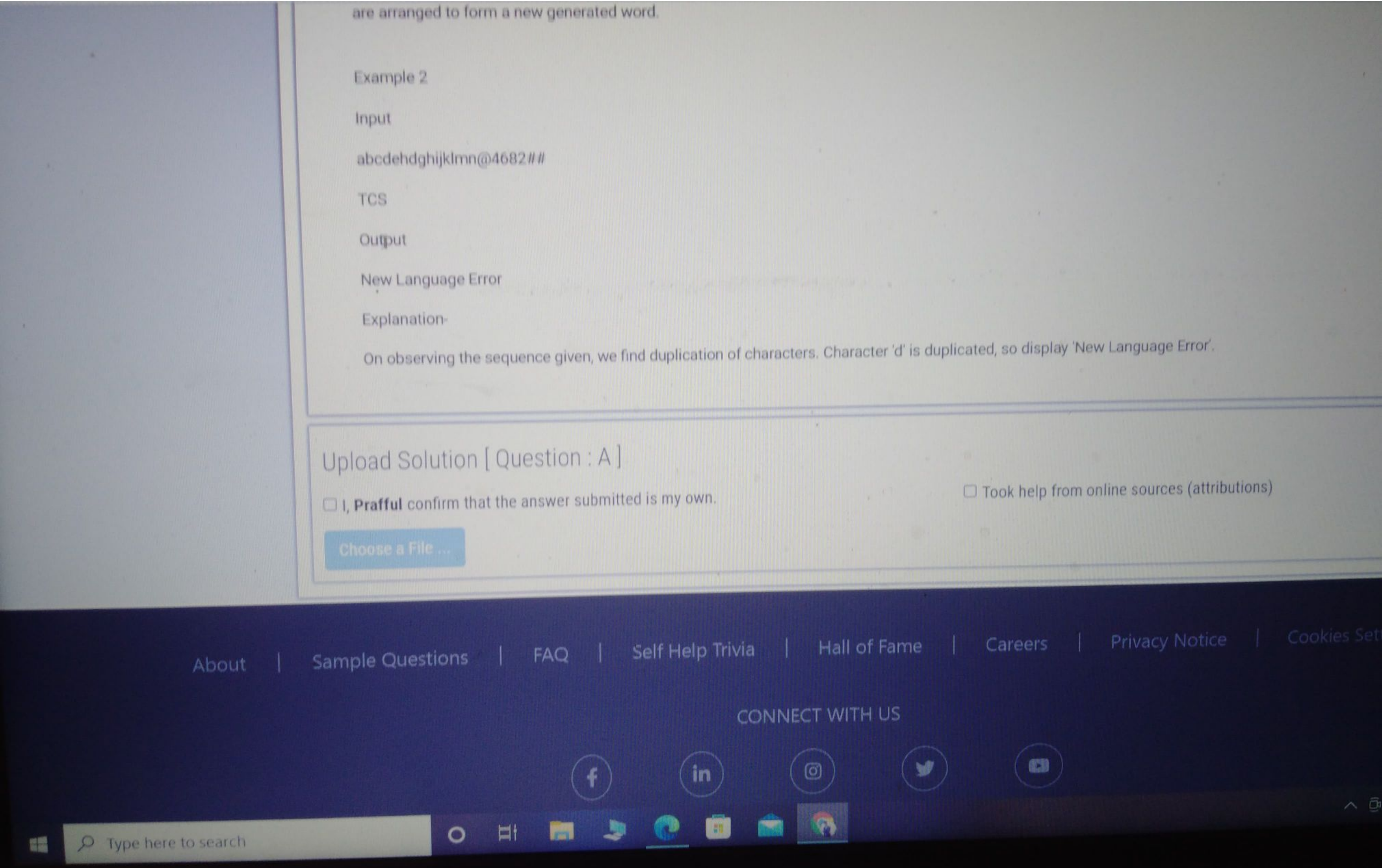
Example 2

Input

```
abcdehdghijklmn@4682##
TCS
```

Output

```
New Language Error
```



are arranged to form a new generated word.

Example 2

Input

abcdehdghijklmn@4682##

TCS

Output

New Language Error

Explanation-

On observing the sequence given, we find duplication of characters. Character 'd' is duplicated, so display 'New Language Error'.

Upload Solution [Question : A]

□ I, **Prafful** confirm that the answer submitted is my own.

□ Took help from online sources (attributions)

Choose a File ...
About | Sample Questions | FAQ | Self Help Trivia | Hall of Fame | Careers | Privacy Notice | Cookies Set
CONNECT WITH US
Type here to search

Polygon 2

− Problem Description

You are given N number of coordinates and your task is to find out how many sides are there in the polygon.

Note: coordinates provided in the input will be in sequential form. Consider that all the coordinates make a close polygon.

− Constraints

2 < N <= 10

− Input

First line contains an integer N which depicts number of co-ordinates.

Next N lines consist of two space separated integers depicting coordinates in the form of x y.

− Output

Print the Number of sides that are there in the polygon.

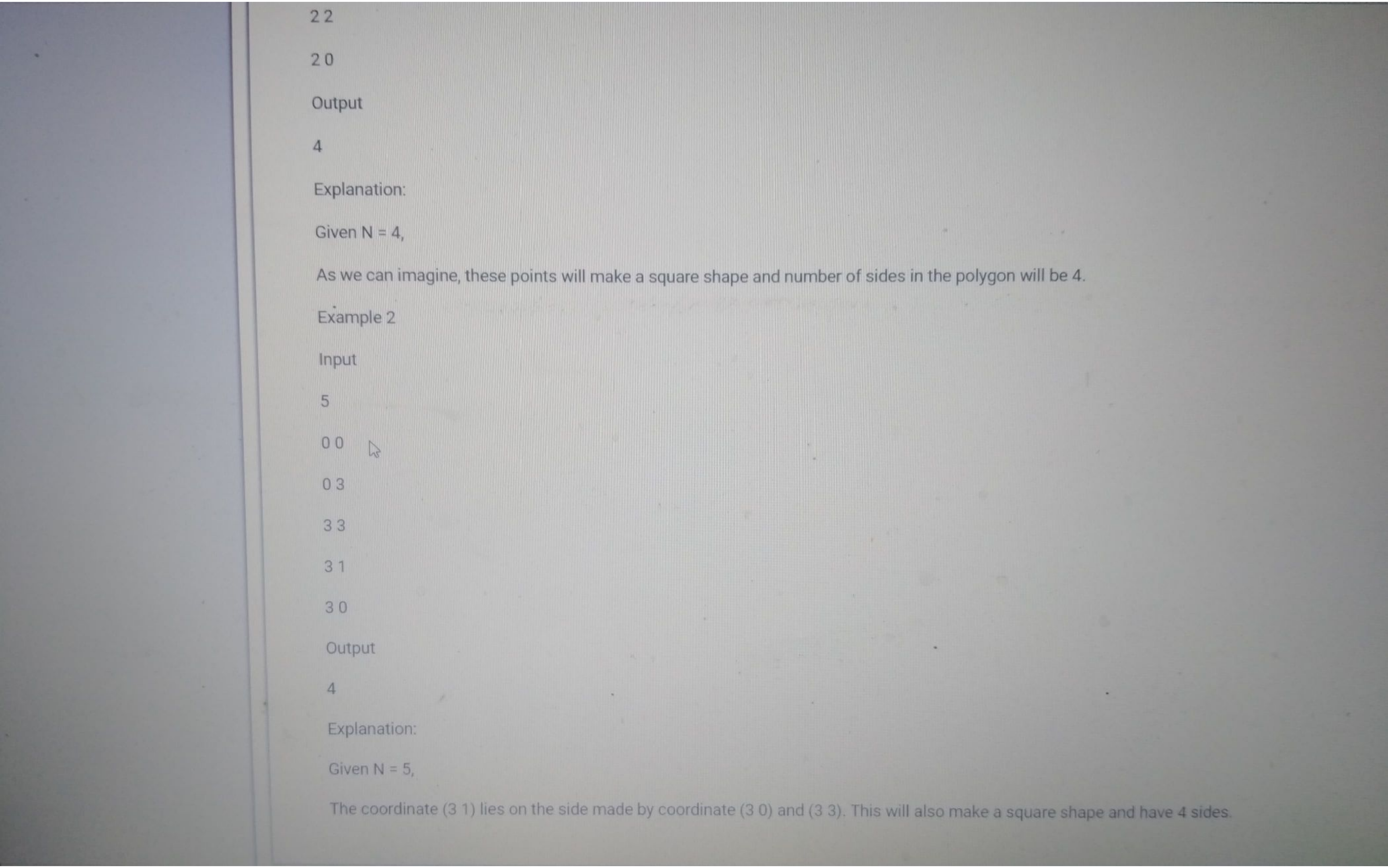
− Time Limit (secs)

1

− Examples

Input

```
4

0 0

0 2

2 2
```



2 2

2 0

Output

4

Explanation:

Given N = 4,

As we can imagine, these points will make a square shape and number of sides in the polygon will be 4.

Example 2

Input

5

0 0

0 3

3 3

3 1

3 0

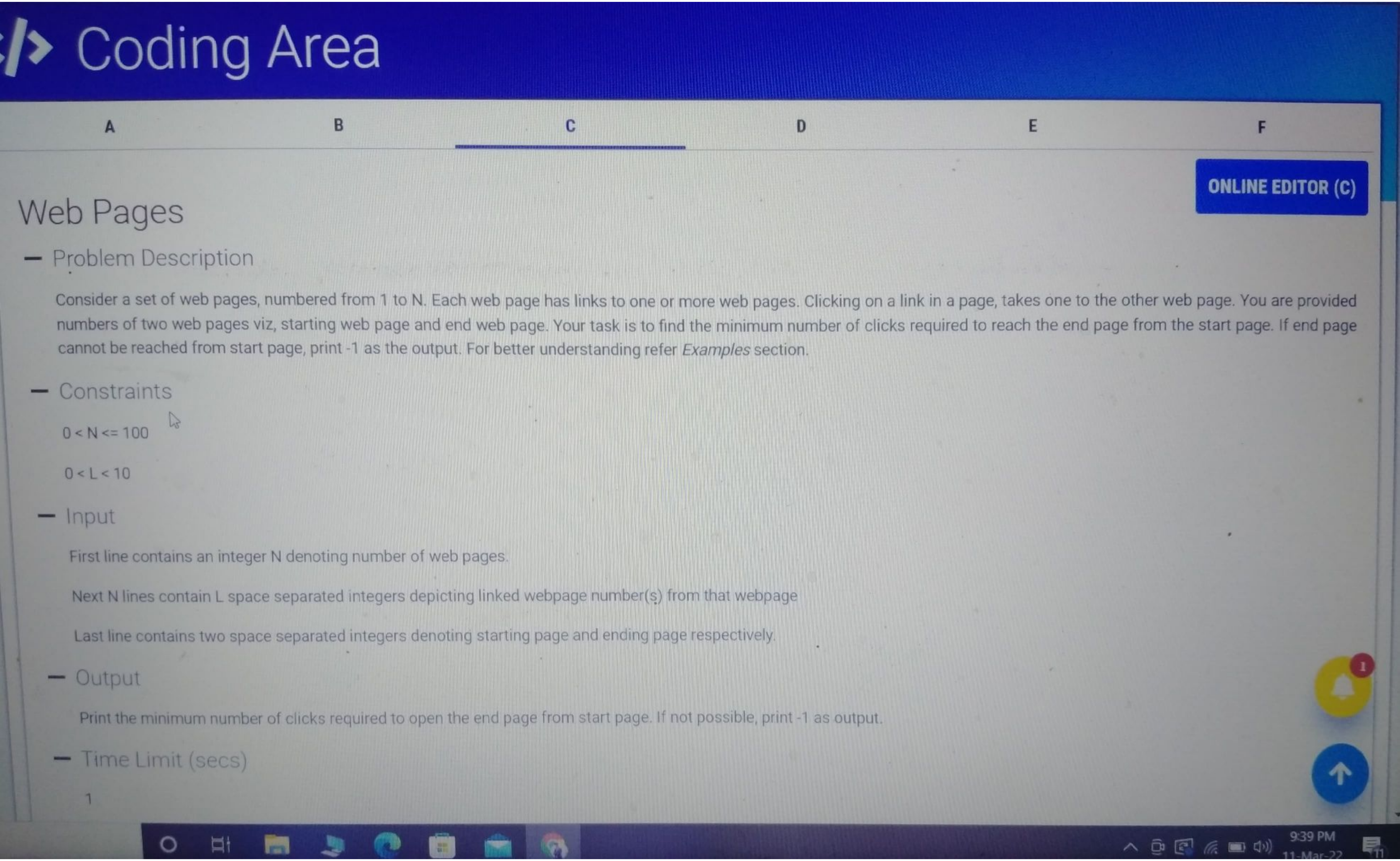
Output

4

Explanation:

Given N = 5,

The coordinate (3 1) lies on the side made by coordinate (3 0) and (3 3). This will also make a square shape and have 4 sides.



Web Pages

— Problem Description

Consider a set of web pages, numbered from 1 to N. Each web page has links to one or more web pages. Clicking on a link in a page, takes one to the other web page. You are provided numbers of two web pages viz, starting web page and end web page. Your task is to find the minimum number of clicks required to reach the end page from the start page. If end page cannot be reached from start page, print -1 as the output. For better understanding refer *Examples* section.

— Constraints

0 < N <= 100

0 < L < 10

— Input

First line contains an integer N denoting number of web pages.

Next N lines contain L space separated integers depicting linked webpage number(s) from that webpage

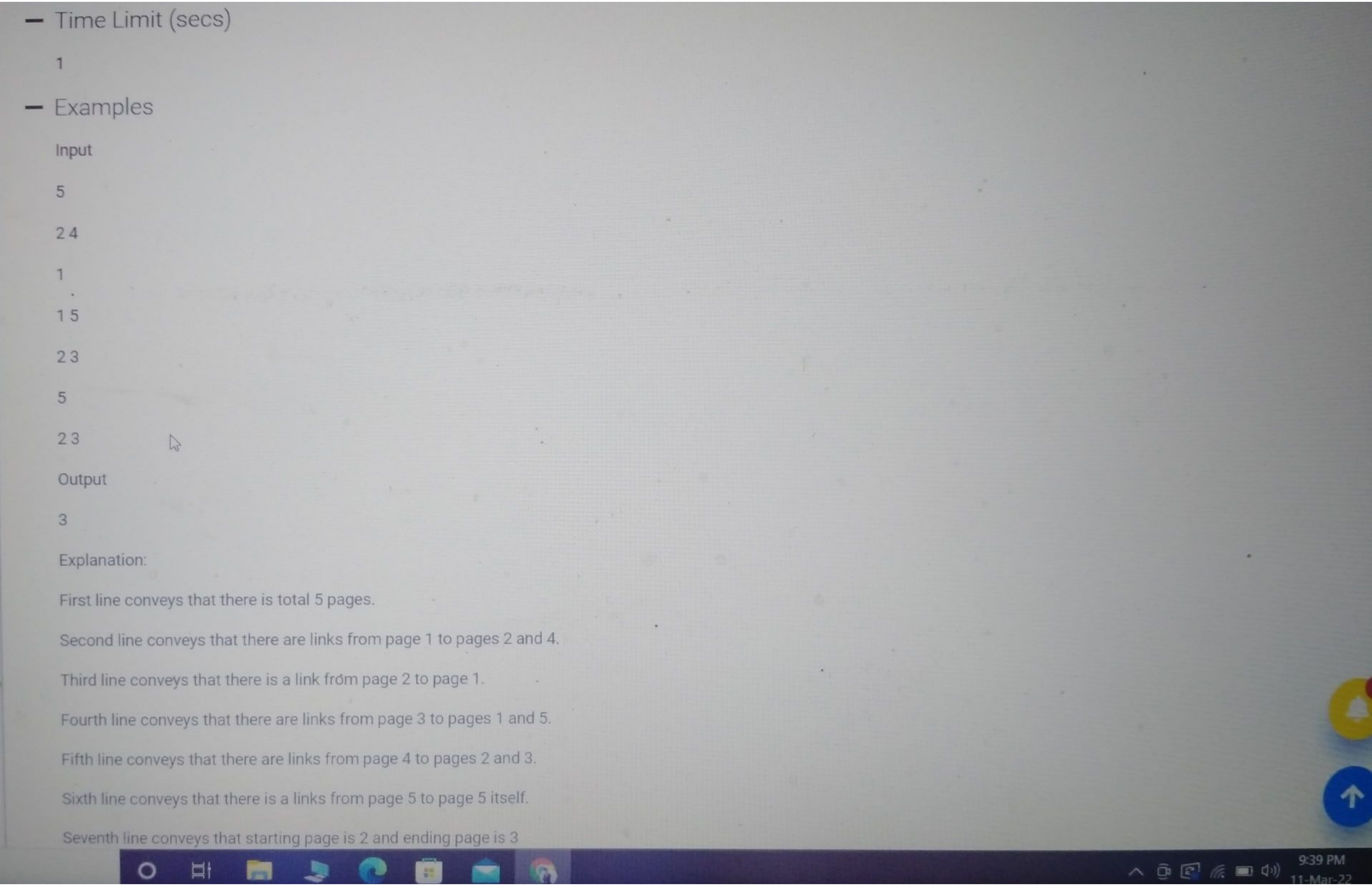
Last line contains two space separated integers denoting starting page and ending page respectively.

— Output

Print the minimum number of clicks required to open the end page from start page. If not possible, print -1 as output.

— Time Limit (secs)

1



Time Limit (secs)

1

Examples

Input

```
5
2 4
1
1 5
2 3
5
2 3
```

Output

```
3
```

Explanation:

First line conveys that there is total 5 pages.

Second line conveys that there are links from page 1 to pages 2 and 4.

Third line conveys that there is a link fróm page 2 to page 1.

Fourth line conveys that there are links from page 3 to pages 1 and 5.

Fifth line conveys that there are links from page 4 to pages 2 and 3.

Sixth line conveys that there is a links from page 5 to page 5 itself.

Seventh line conveys that starting page is 2 and ending page is 3

Sixth line conveys that there is a links from page 5 to page 5 itself.

Seventh line conveys that starting page is 2 and ending page is 3

From page 2, we can open only page 1. From page 1, we can open page 4. From page 4, we can open page 3. So, minimum 3 clicks are required, and this is the output.

Example 2

Input

3

2

1

1

2 3

Output

-1

Explanation:

First line conveys that there is total 3 pages.

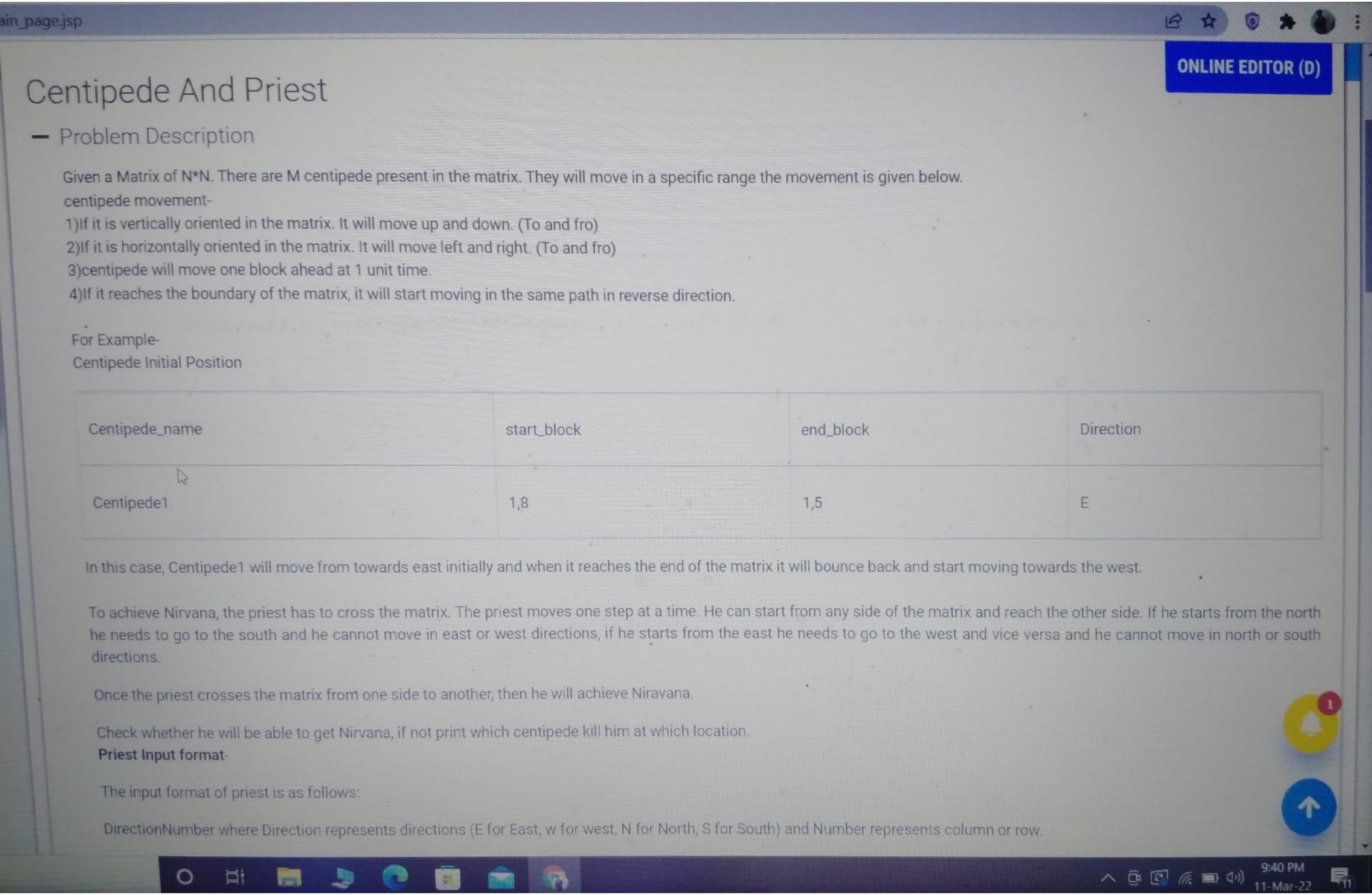
Second line conveys that there are links from page 1 to page 2.

Third line conveys that there is a link from page 2 to page 1.

Fourth line conveys that there are links from page 3 to page 1.

Since there is no way to reach from page 2 to page 3, print -1 as output.

Upload Solution [Question : C]



Centipede And Priest

— Problem Description

Given a Matrix of N*N. There are M centipede present in the matrix. They will move in a specific range the movement is given below.
centipede movement-
1)If it is vertically oriented in the matrix. It will move up and down. (To and fro)
2)If it is horizontally oriented in the matrix. It will move left and right. (To and fro)
3)centipede will move one block ahead at 1 unit time.
4)If it reaches the boundary of the matrix, it will start moving in the same path in reverse direction.

For Example-
Centipede Initial Position

Centipede_name	start_block	end_block	Direction
Centipede1	1,8	1,5	E

In this case, Centipede1 will move from towards east initially and when it reaches the end of the matrix it will bounce back and start moving towards the west.

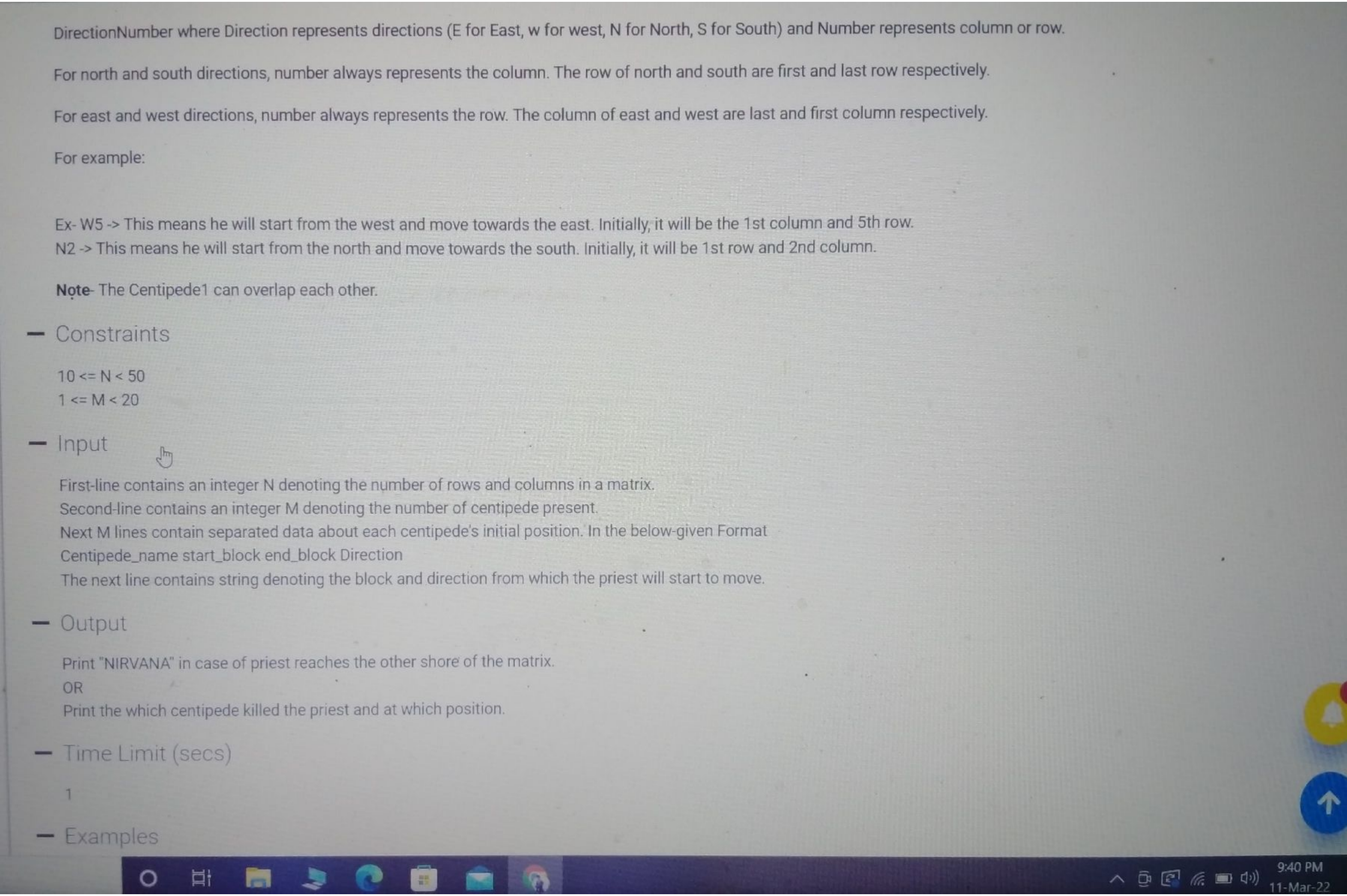
To achieve Nirvana, the priest has to cross the matrix. The priest moves one step at a time. He can start from any side of the matrix and reach the other side. If he starts from the north he needs to go to the south and he cannot move in east or west directions, if he starts from the east he needs to go to the west and vice versa and he cannot move in north or south directions.

Once the priest crosses the matrix from one side to another, then he will achieve Niravana.

Check whether he will be able to get Nirvana, if not print which centipede kill him at which location.
Priest Input format-

The input format of priest is as follows:

DirectionNumber where Direction represents directions (E for East, w for west, N for North, S for South) and Number represents column or row.



DirectionNumber where Direction represents directions (E for East, w for west, N for North, S for South) and Number represents column or row.

For north and south directions, number always represents the column. The row of north and south are first and last row respectively.

For east and west directions, number always represents the row. The column of east and west are last and first column respectively.

For example:

Ex- W5 -> This means he will start from the west and move towards the east. Initially, it will be the 1st column and 5th row.
N2 -> This means he will start from the north and move towards the south. Initially, it will be 1st row and 2nd column.

Note- The Centipede1 can overlap each other.

Constraints

10 <= N < 50
1 <= M < 20

Input

First-line contains an integer N denoting the number of rows and columns in a matrix.
Second-line contains an integer M denoting the number of centipede present.
Next M lines contain separated data about each centipede's initial position. In the below-given Format
Centipede_name start_block end_block Direction
The next line contains string denoting the block and direction from which the priest will start to move.

Output

Print "NIRVANA" in case of priest reaches the other shore of the matrix.
OR
Print the which centipede killed the priest and at which position.

Time Limit (secs)

1

Examples

– Examples

Input

```
10
4
Centipede1 1,5 1,8 E
Centipede2 7,4 7,7 W
Centipede3 2,10 4,10 N
Centipede4 2,2 4,2 N
W2
```

Output

```
Centipede4 2,2
```

Explanation-

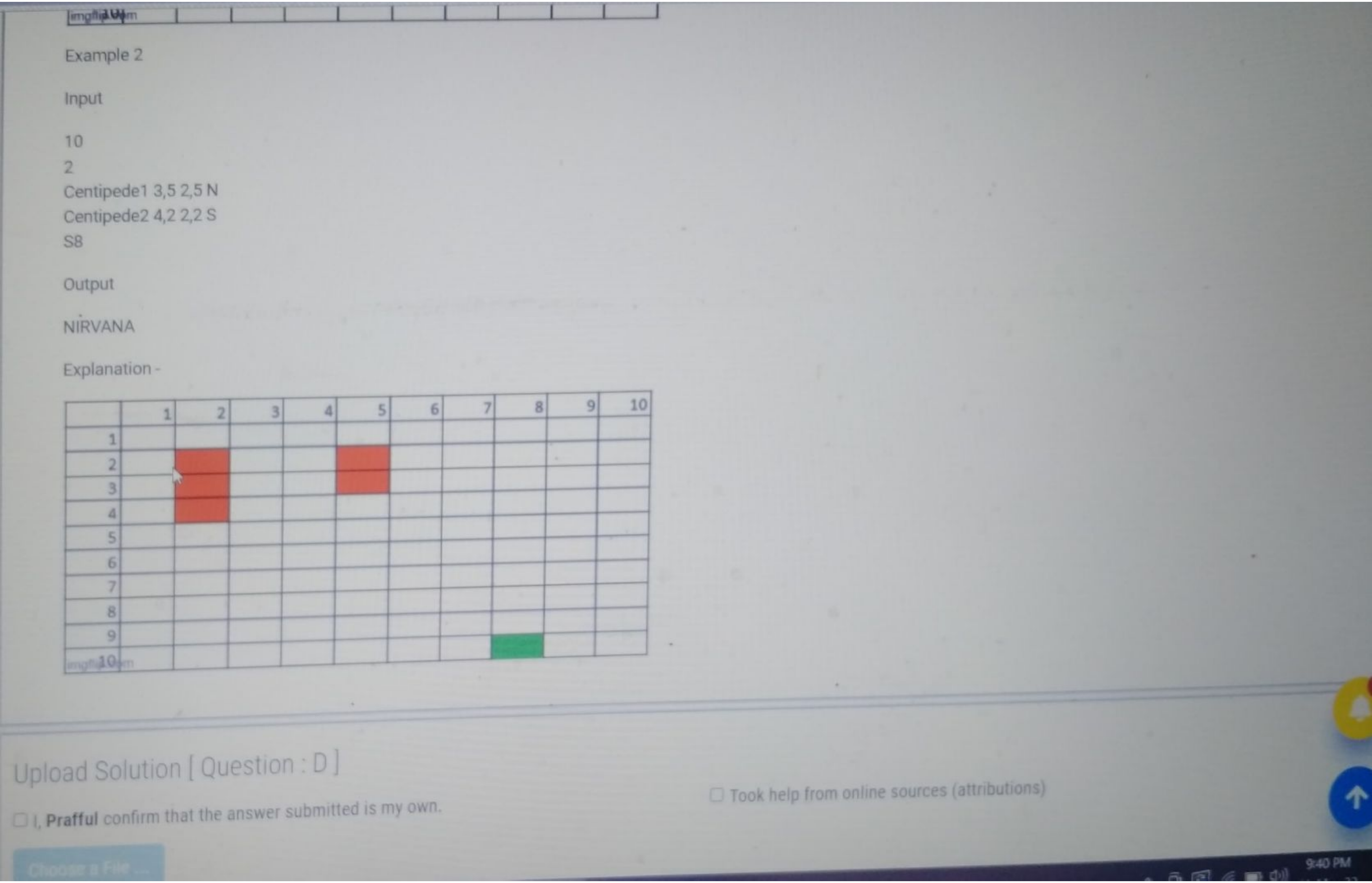
🟥 Centipede

🟩 Priest

⬛ centipede and priest

	1	2	3	4	5	6	7	8	9	10
1		🟥				🟥	🟥	🟥	🟥	🟥
2		⬛								🟥
3		🟥								🟥
4										
5										
6										
7			🟥	🟥	🟥	🟥				
8										
9										
10										

Example 2



10										

Example 2

Input

```
10
2
Centipede1 3,5 2,5 N
Centipede2 4,2 2,2 S
S8
```

Output

```
NIRVANA
```

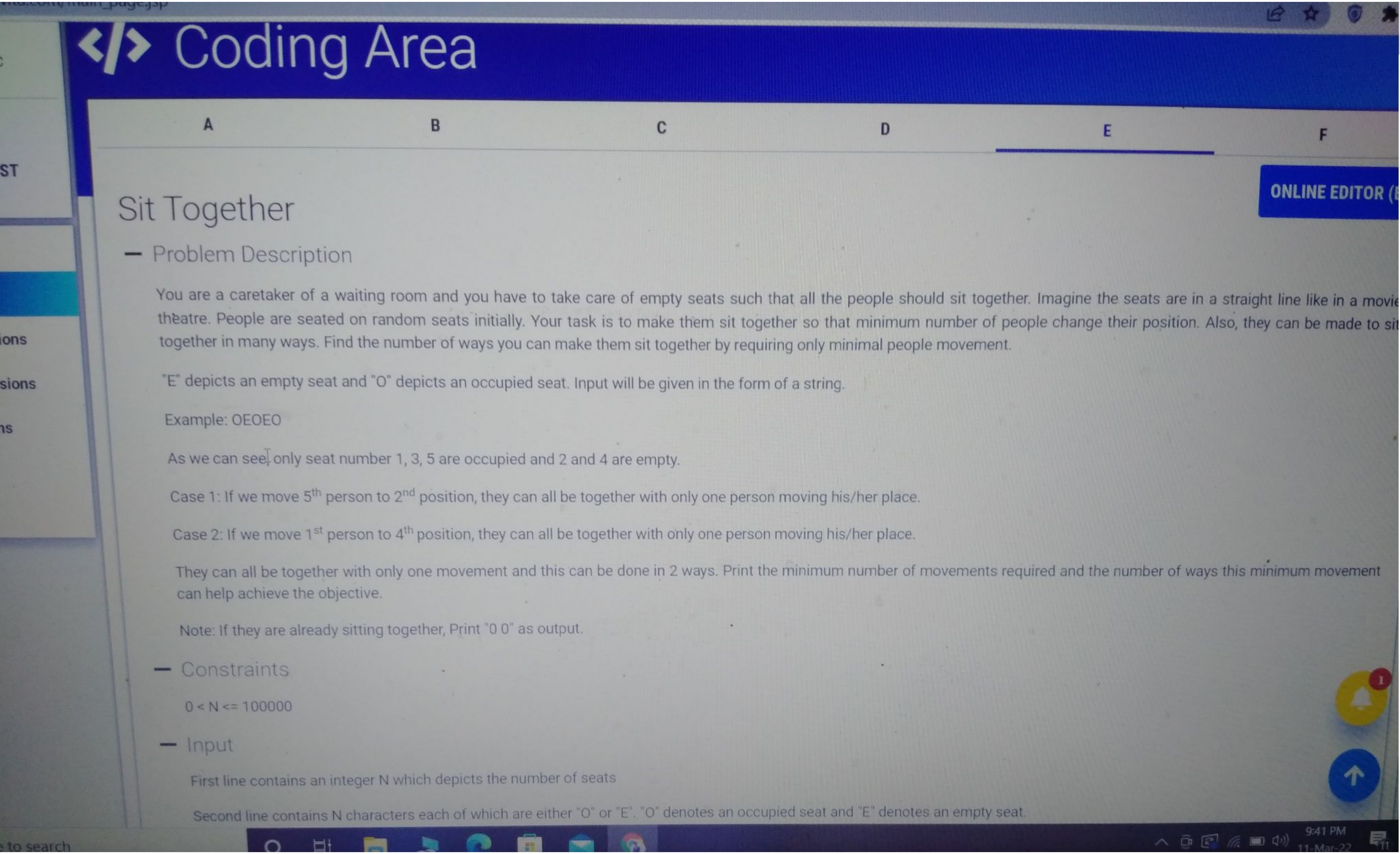
Explanation -

	1	2	3	4	5	6	7	8	9	10
1										
2		🟥			🟥					
3		🟥			🟥					
4		🟥								
5										
6										
7										
8										
9										
10								🟩		

Upload Solution [Question : D]

□ I, **Prafful** confirm that the answer submitted is my own.

□ Took help from online sources (attributions)



</> Coding Area

Sit Together

— Problem Description

You are a caretaker of a waiting room and you have to take care of empty seats such that all the people should sit together. Imagine the seats are in a straight line like in a movi theatre. People are seated on random seats initially. Your task is to make them sit together so that minimum number of people change their position. Also, they can be made to si together in many ways. Find the number of ways you can make them sit together by requiring only minimal people movement.

"E" depicts an empty seat and "O" depicts an occupied seat. Input will be given in the form of a string.

Example: OEOEO

As we can see, only seat number 1, 3, 5 are occupied and 2 and 4 are empty.

Case 1: If we move 5th person to 2nd position, they can all be together with only one person moving his/her place.

Case 2: If we move 1st person to 4th position, they can all be together with only one person moving his/her place.

They can all be together with only one movement and this can be done in 2 ways. Print the minimum number of movements required and the number of ways this minimum movement can help achieve the objective.

Note: If they are already sitting together, Print "0 0" as output.

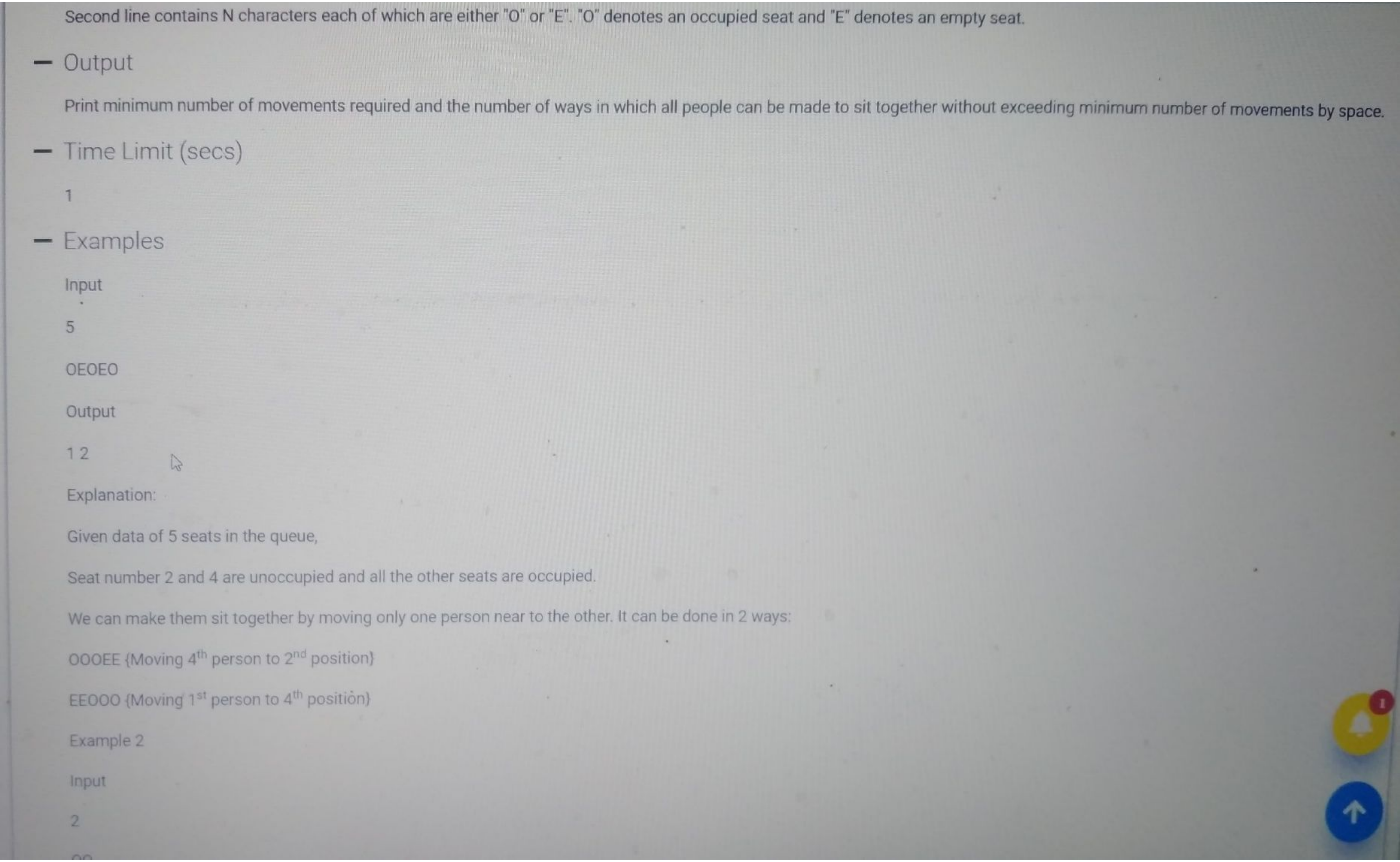
— Constraints

0 < N <= 100000

— Input

First line contains an integer N which depicts the number of seats

Second line contains N characters each of which are either "O" or "E". "O" denotes an occupied seat and "E" denotes an empty seat.



Second line contains N characters each of which are either "O" or "E". "O" denotes an occupied seat and "E" denotes an empty seat.

Output

Print minimum number of movements required and the number of ways in which all people can be made to sit together without exceeding minimum number of movements by space.

Time Limit (secs)

1

Examples

Input

5

OEOEO

Output

1 2

Explanation:

Given data of 5 seats in the queue,

Seat number 2 and 4 are unoccupied and all the other seats are occupied.

We can make them sit together by moving only one person near to the other. It can be done in 2 ways:

OOOEE {Moving 4th person to 2nd position}

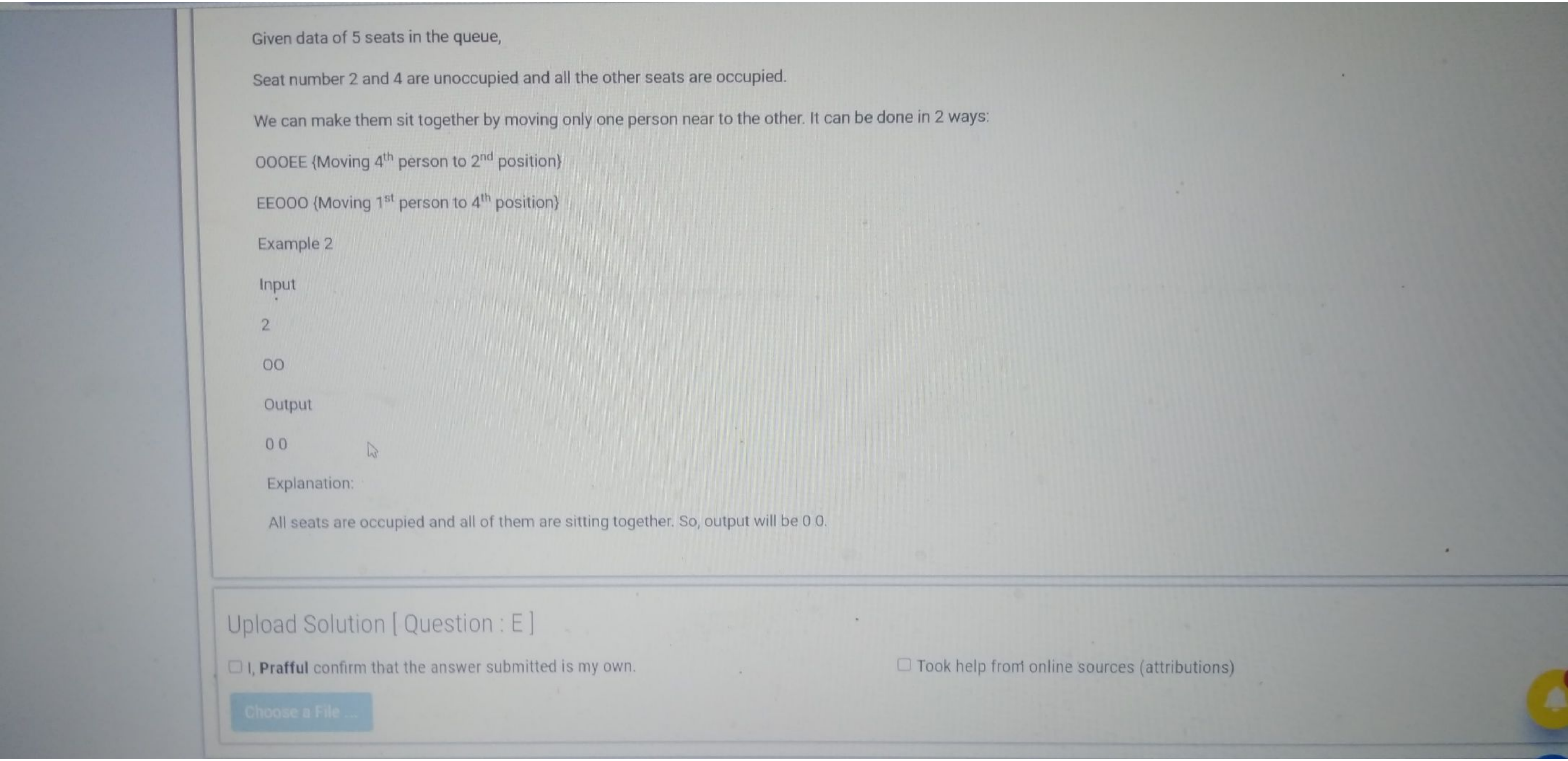
EEOOO {Moving 1st person to 4th position}

Example 2

Input

2

OO



Given data of 5 seats in the queue,

Seat number 2 and 4 are unoccupied and all the other seats are occupied.

We can make them sit together by moving only one person near to the other. It can be done in 2 ways:

OOOEE {Moving 4th person to 2nd position}

EEOOO {Moving 1st person to 4th position}

Example 2

Input

2

OO

Output

0 0

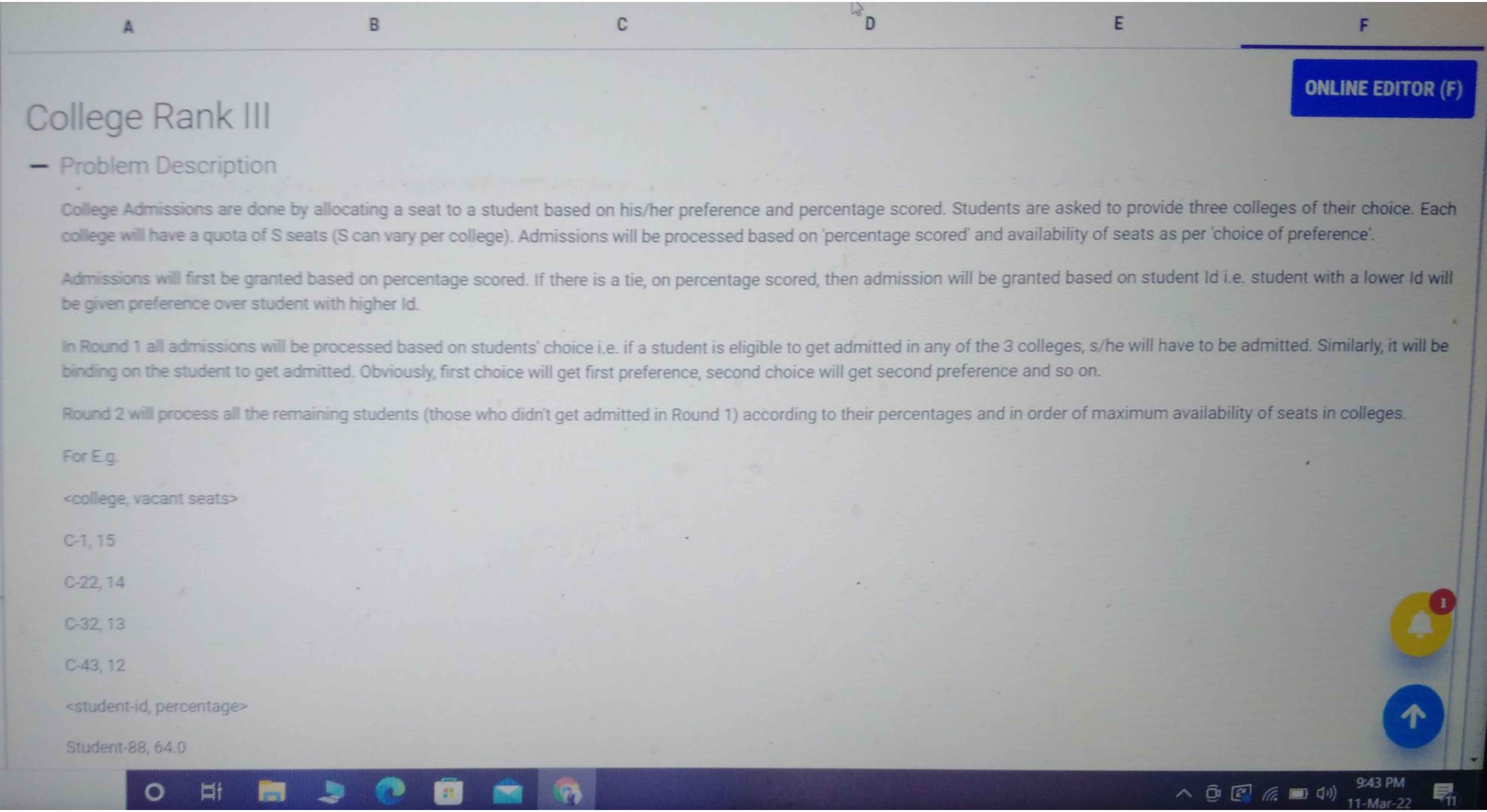
Explanation:

All seats are occupied and all of them are sitting together. So, output will be 0 0.

Upload Solution [Question : E]

☐ I, **Prafful** confirm that the answer submitted is my own.

☐ Took help from online sources (attributions)



College Rank III

– Problem Description

College Admissions are done by allocating a seat to a student based on his/her preference and percentage scored. Students are asked to provide three colleges of their choice. Each college will have a quota of S seats (S can vary per college). Admissions will be processed based on 'percentage scored' and availability of seats as per 'choice of preference'.

Admissions will first be granted based on percentage scored. If there is a tie, on percentage scored, then admission will be granted based on student Id i.e. student with a lower Id will be given preference over student with higher Id.

In Round 1 all admissions will be processed based on students' choice i.e. if a student is eligible to get admitted in any of the 3 colleges, s/he will have to be admitted. Similarly, it will be binding on the student to get admitted. Obviously, first choice will get first preference, second choice will get second preference and so on.

Round 2 will process all the remaining students (those who didn't get admitted in Round 1) according to their percentages and in order of maximum availability of seats in colleges.

For E.g.

<college, vacant seats>

C-1, 15

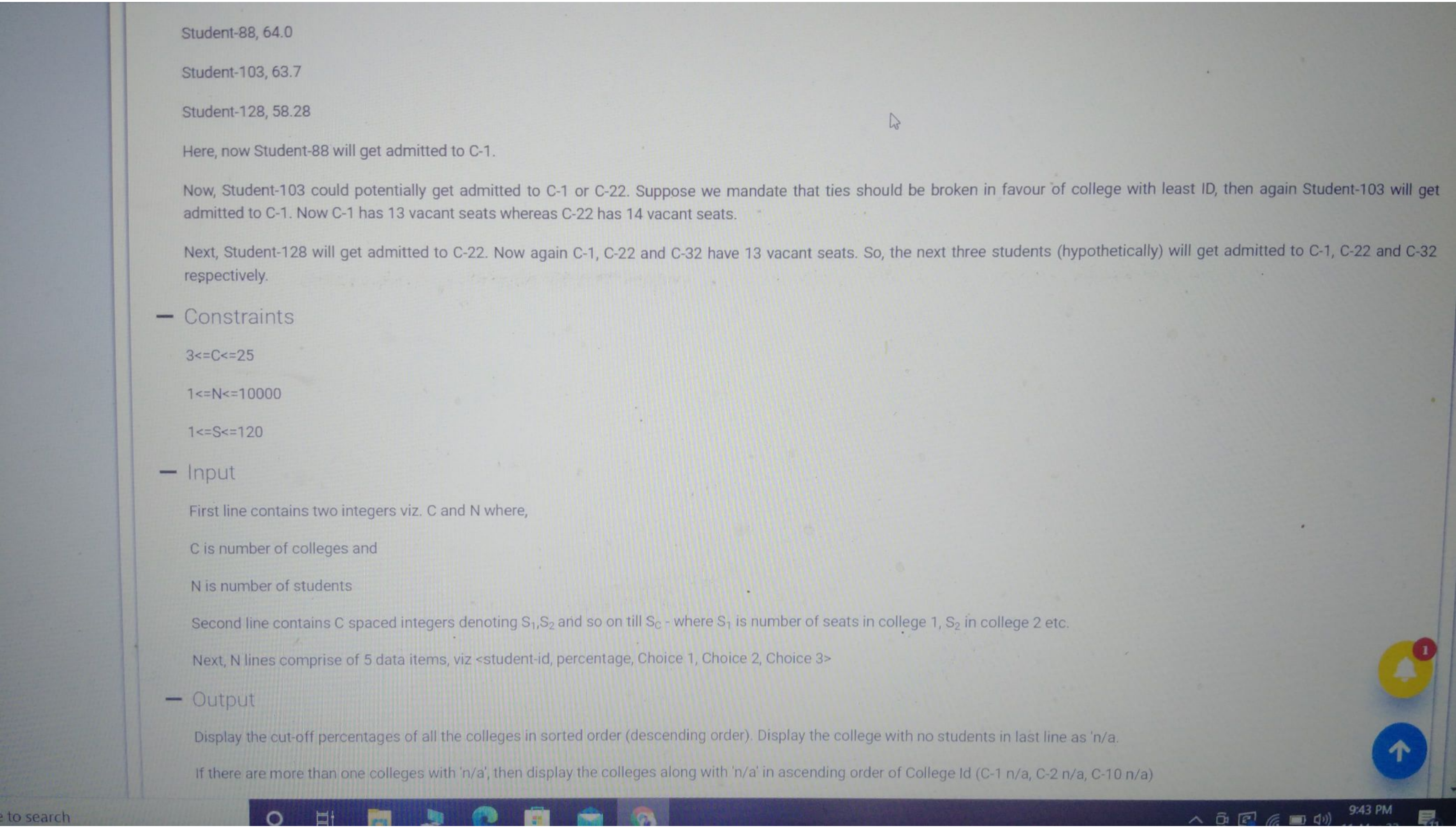
C-22, 14

C-32, 13

C-43, 12

<student-id, percentage>

Student-88, 64.0



Student-88, 64.0

Student-103, 63.7

Student-128, 58.28

Here, now Student-88 will get admitted to C-1.

Now, Student-103 could potentially get admitted to C-1 or C-22. Suppose we mandate that ties should be broken in favour of college with least ID, then again Student-103 will get admitted to C-1. Now C-1 has 13 vacant seats whereas C-22 has 14 vacant seats.

Next, Student-128 will get admitted to C-22. Now again C-1, C-22 and C-32 have 13 vacant seats. So, the next three students (hypothetically) will get admitted to C-1, C-22 and C-32 respectively.

— Constraints

3<=C<=25

1<=N<=10000

1<=S<=120

— Input

First line contains two integers viz. C and N where,

C is number of colleges and

N is number of students

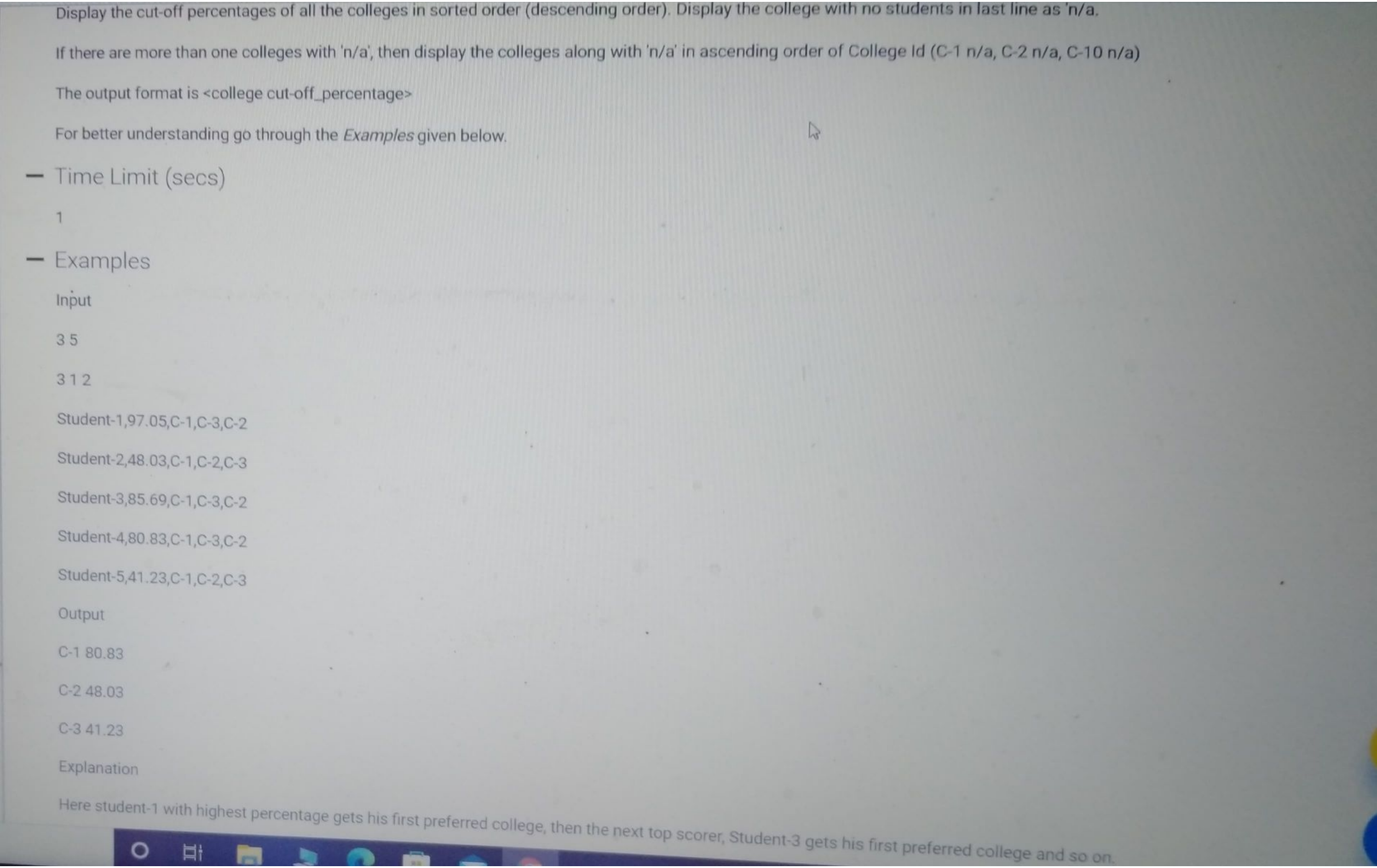
Second line contains C spaced integers denoting S_1,S_2 and so on till S_C - where S_1 is number of seats in college 1, S_2 in college 2 etc.

Next, N lines comprise of 5 data items, viz <student-id, percentage, Choice 1, Choice 2, Choice 3>

— Output

Display the cut-off percentages of all the colleges in sorted order (descending order). Display the college with no students in last line as 'n/a.

If there are more than one colleges with 'n/a', then display the colleges along with 'n/a' in ascending order of College Id (C-1 n/a, C-2 n/a, C-10 n/a)



Display the cut-off percentages of all the colleges in sorted order (descending order). Display the college with no students in last line as 'n/a.

If there are more than one colleges with 'n/a', then display the colleges along with 'n/a' in ascending order of College Id (C-1 n/a, C-2 n/a, C-10 n/a)

The output format is <college cut-off_percentage>

For better understanding go through the *Examples* given below.

Time Limit (secs)

1

Examples

Input

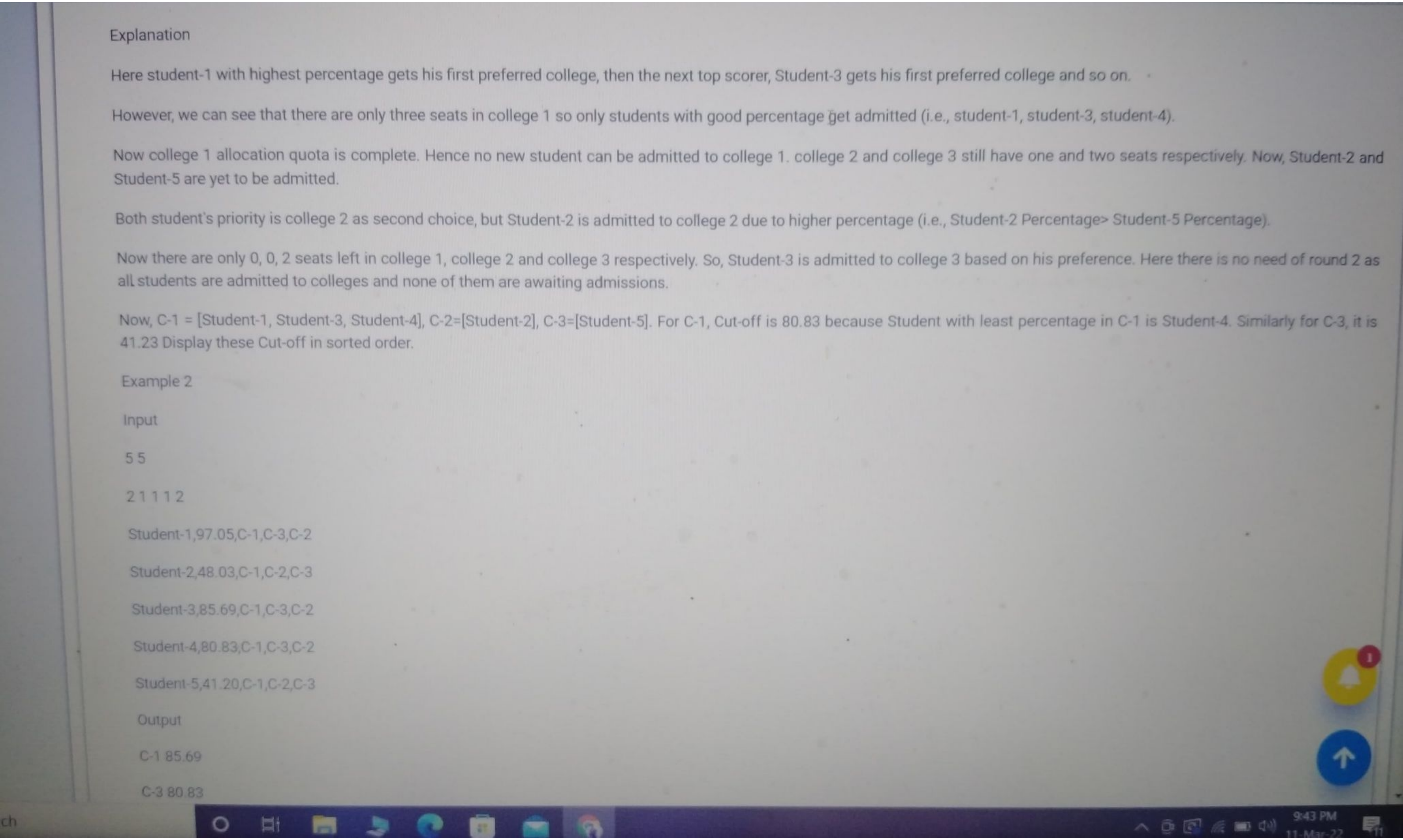
```
3 5
3 1 2
Student-1,97.05,C-1,C-3,C-2
Student-2,48.03,C-1,C-2,C-3
Student-3,85.69,C-1,C-3,C-2
Student-4,80.83,C-1,C-3,C-2
Student-5,41.23,C-1,C-2,C-3
```

Output

```
C-1 80.83
C-2 48.03
C-3 41.23
```

Explanation

Here student-1 with highest percentage gets his first preferred college, then the next top scorer, Student-3 gets his first preferred college and so on.



Explanation

Here student-1 with highest percentage gets his first preferred college, then the next top scorer, Student-3 gets his first preferred college and so on.

However, we can see that there are only three seats in college 1 so only students with good percentage get admitted (i.e., student-1, student-3, student-4).

Now college 1 allocation quota is complete. Hence no new student can be admitted to college 1. college 2 and college 3 still have one and two seats respectively. Now, Student-2 and Student-5 are yet to be admitted.

Both student's priority is college 2 as second choice, but Student-2 is admitted to college 2 due to higher percentage (i.e., Student-2 Percentage> Student-5 Percentage).

Now there are only 0, 0, 2 seats left in college 1, college 2 and college 3 respectively. So, Student-3 is admitted to college 3 based on his preference. Here there is no need of round 2 as all students are admitted to colleges and none of them are awaiting admissions.

Now, C-1 = [Student-1, Student-3, Student-4], C-2=[Student-2], C-3=[Student-5]. For C-1, Cut-off is 80.83 because Student with least percentage in C-1 is Student-4. Similarly for C-3, it is 41.23 Display these Cut-off in sorted order.

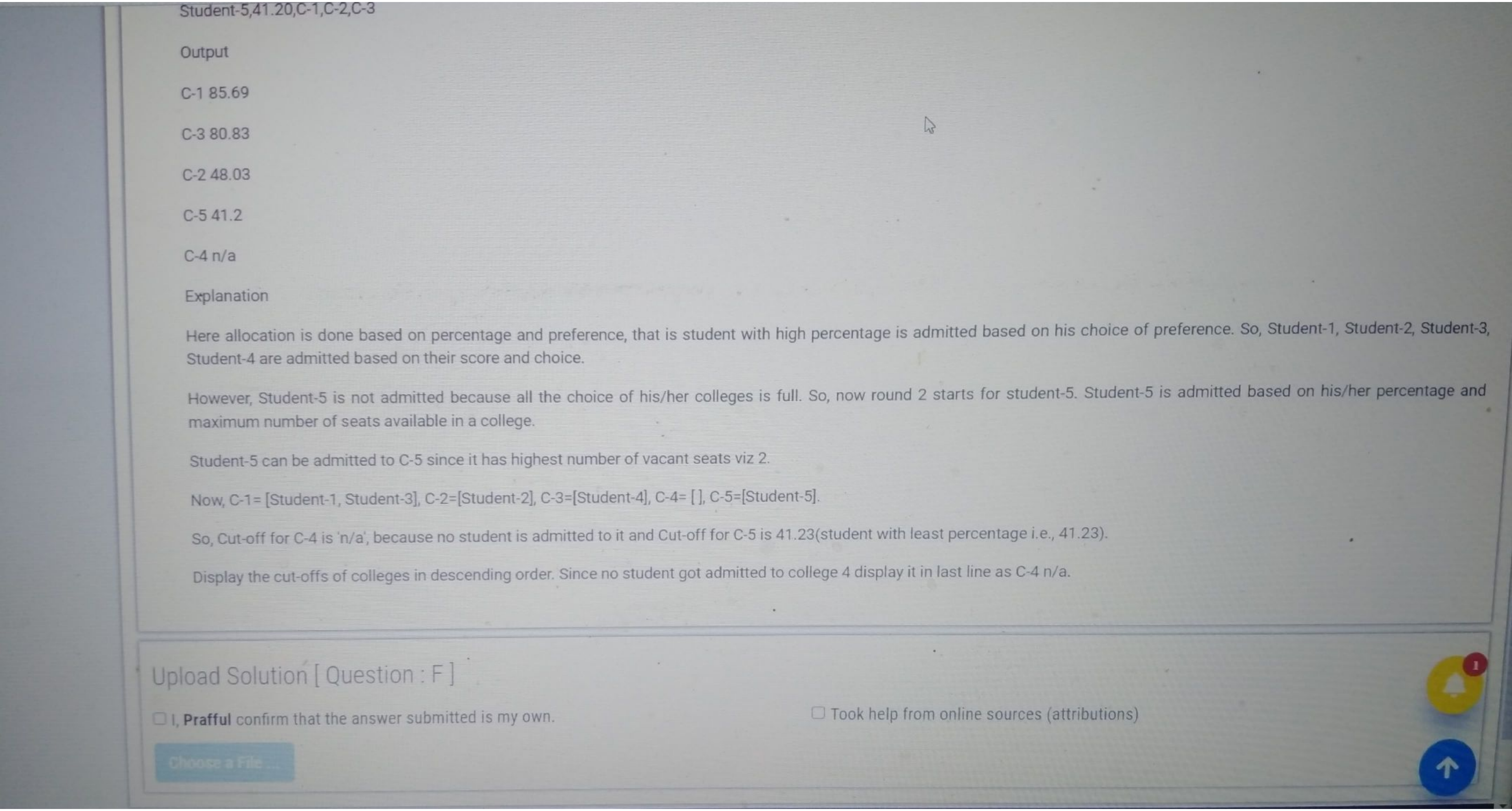
Example 2

Input

```
5 5

2 1 1 1 2

Student-1,97.05,C-1,C-3,C-2

Student-2,48.03,C-1,C-2,C-3

Student-3,85.69,C-1,C-3,C-2

Student-4,80.83,C-1,C-3,C-2

Student-5,41.20,C-1,C-2,C-3
```

Output

```
C-1 85.69

C-3 80.83
```



```
Student-5,41.20,C-1,C-2,C-3
```

Output

```
C-1 85.69

C-3 80.83

C-2 48.03

C-5 41.2

C-4 n/a
```

Explanation

Here allocation is done based on percentage and preference, that is student with high percentage is admitted based on his choice of preference. So, Student-1, Student-2, Student-3, Student-4 are admitted based on their score and choice.

However, Student-5 is not admitted because all the choice of his/her colleges is full. So, now round 2 starts for student-5. Student-5 is admitted based on his/her percentage and maximum number of seats available in a college.

Student-5 can be admitted to C-5 since it has highest number of vacant seats viz 2.

Now, C-1= [Student-1, Student-3], C-2=[Student-2], C-3=[Student-4], C-4= [ ], C-5=[Student-5].

So, Cut-off for C-4 is 'n/a', because no student is admitted to it and Cut-off for C-5 is 41.23(student with least percentage i.e., 41.23).

Display the cut-offs of colleges in descending order. Since no student got admitted to college 4 display it in last line as C-4 n/a.

Upload Solution [Question : F]

□ I, **Prafful** confirm that the answer submitted is my own.

□ Took help from online sources (attributions)