Idea

We understood the problem 1 as a given graph where we need to find the groups that this graph has. A group is formed when a user (vertex) or a group of users have no connections with another user or group of users.

Methodology

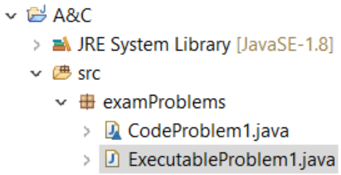
We used the Depth-First search algorithm in order to do the problem. The given graph formed a tree. We went through all the users one by one, exploring as far as possible each branch of the tree before doing backtracking. If a user “u1” was connected to a user “u2” we no longer needed to do backtracking on “u2” as it formed part of the “u1” group (it was a candidate for u1). After the first group is closed, we just started again with a user that is not part of another group (known as non-visited user).

Following the theory:

- If the condition is not met, the last chosen candidate is rejected, and it is not considered again in the future for the given group.
- If it is condition is met, the candidate is incorporated into the group of chosen ones and always remains in it.

Execution

The problem is divided into the code and the executable.

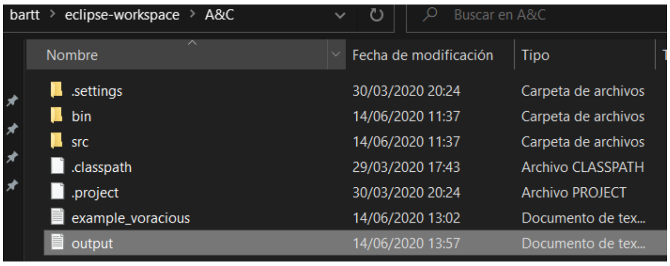


```
A&C
  JRE System Library [JavaSE-1.8]
  src
    examProblems
      CodeProblem1.java
      ExecutableProblem1.java
```

We just need to open the executable and insert the path of the input file (it should be in the workspace)

```
Insert the path of the “example_voracious.txt” file:
example_voracious.txt
```

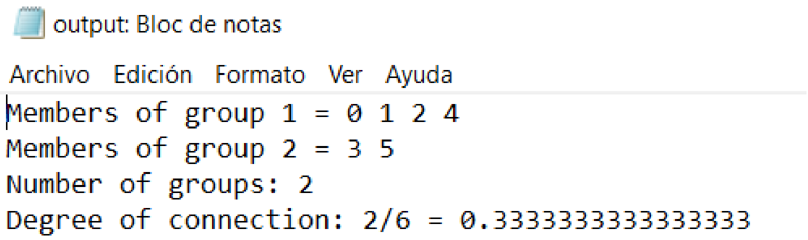
Like that an output file will appear in the workspace:



Nombre	Fecha de modificación	Tipo
.settings	30/03/2020 20:24	Carpeta de archivos
bin	14/06/2020 11:37	Carpeta de archivos
src	14/06/2020 11:37	Carpeta de archivos
.classpath	29/03/2020 17:43	Archivo CLASSPATH
.project	30/03/2020 20:24	Archivo PROJECT
example_voracious	14/06/2020 13:02	Documento de tex...
output	14/06/2020 13:57	Documento de tex...

Output file

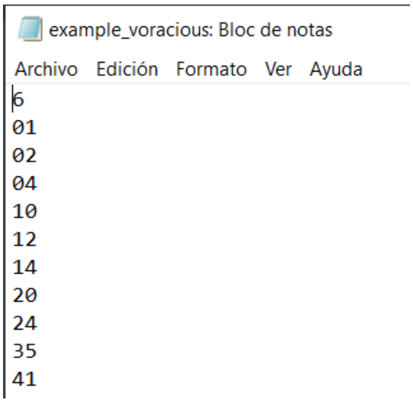
The output file contains the solution of the problem. Following the given example, we obtain the following output:



```
Members of group 1 = 0 1 2 4
Members of group 2 = 3 5
Number of groups: 2
Degree of connection: 2/6 = 0.3333333333333333
```

Input File

The first line indicates the number of users. The following lines indicate the connection between the source user (first number in the line) and the destination user (second number in the line).


example_voracious: Bloc de notas
Archivo Edición Formato Ver Ayuda

```
6
01
02
04
10
12
14
20
24
35
41
```

In this example we have 6 users. The first connection is User0 with User1, the second connection is User0 with User2.

We used the data given in problem1´s example. However, we used numbers instead of names (this method is easier and does not interfere with the algorithm, which is the important part of the exercise).