CS550: Massive Data Mining and Learning Spring 2021

Problem Set 1

Due 11:59pm Friday, Feb. 19, 2021

Only one late period is allowed for this homework (11:59pm Saturday 2/20)

**Submission Instructions**

**Assignment Submission**: Include a signed agreement to the Honor Code with this assignment. Assignments are due at 11:59pm. All students must submit their homework via Canvas. Students can typeset or scan their homework. Students also need to include their code in the final submission zip file. Put all the code for a single question into a single file.

**Late Day Policy**: Each student will have a total of **two** free late days, and for each homework only one late day can be used. If a late day is used, the due date is 11:59pm on the next day.

**Honor Code**: Students may discuss and work on homework problems in groups. This is encouraged. However, each student must write down their solutions independently to show they understand the solution well enough in order to reconstruct it by themselves. Students should clearly mention the names of all the other students who were part of their discussion group. Using code or solutions obtained from the web is considered an honor code violation. We check all the submissions for plagiarism. We take the honor code seriously and expect students to do the same.

Discussion Group (People with whom you discussed ideas used in your answers):

1. Kaushal Patel
2. Chaitanya

On-line or hardcopy documents used as part of your answers:

I acknowledge and accept the Honor Code.

(Signed)\_\_Viren Patel\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

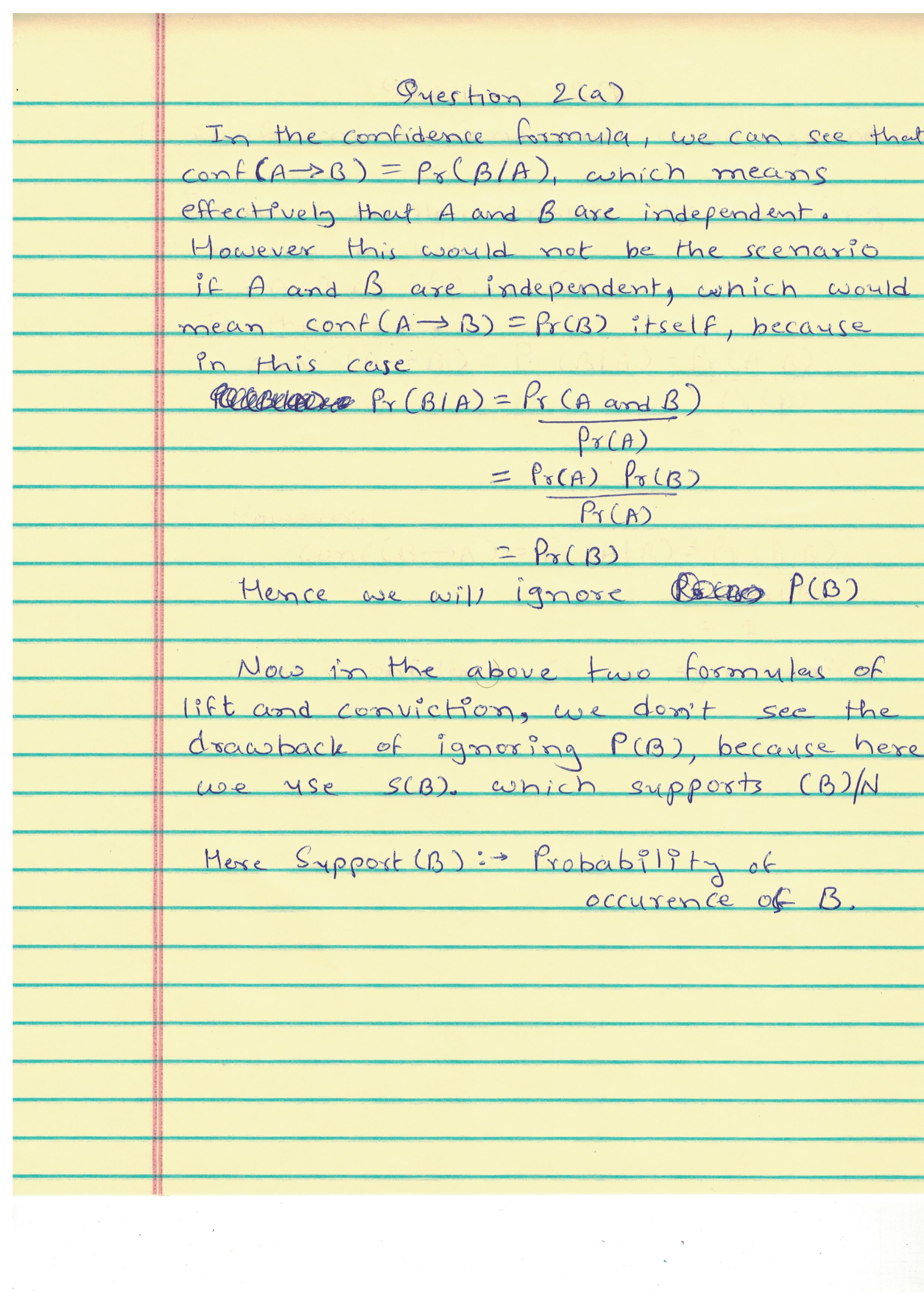
If you are not printing this document out, please type your initials above.

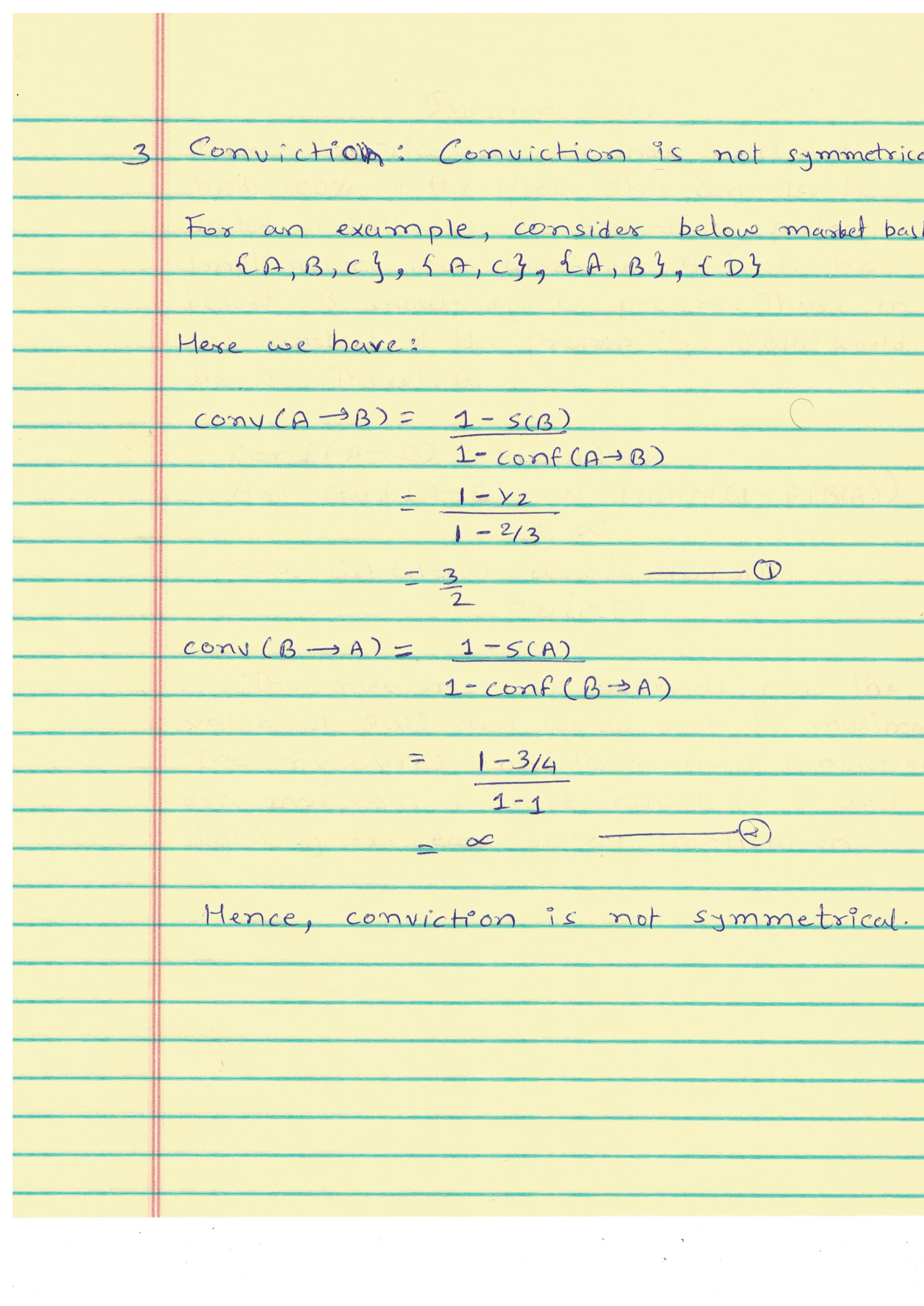
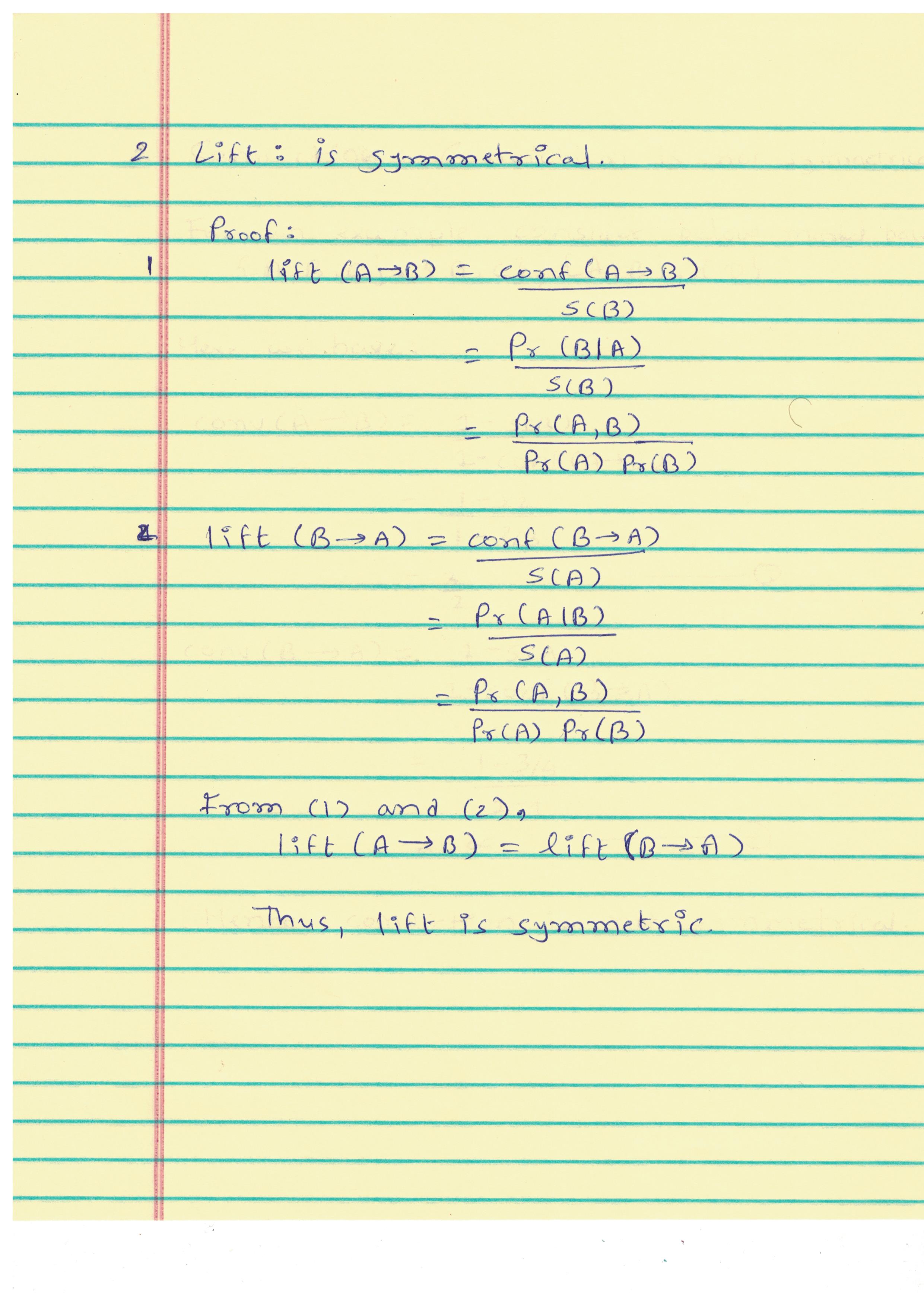
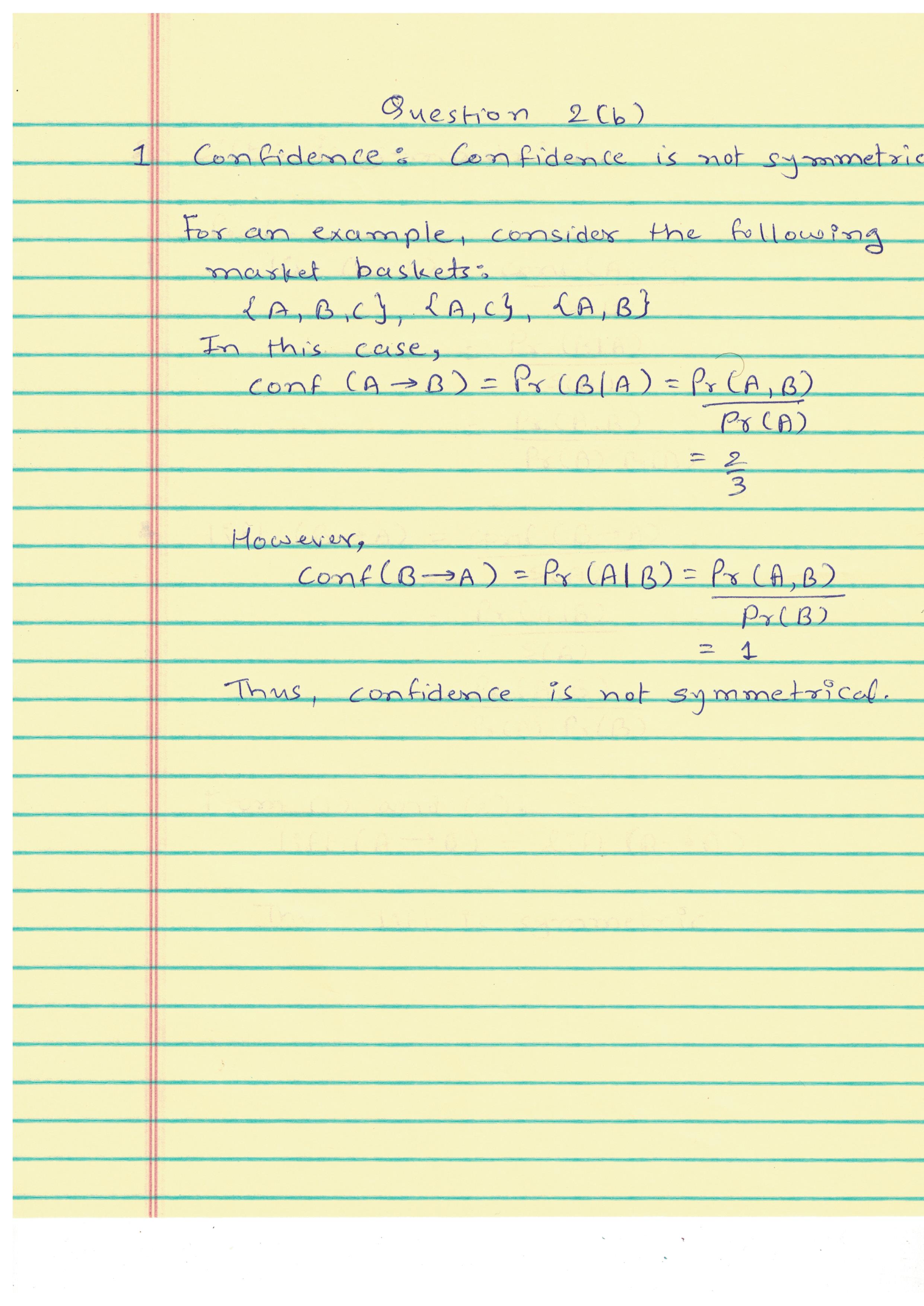
**Answer to Questions 1**

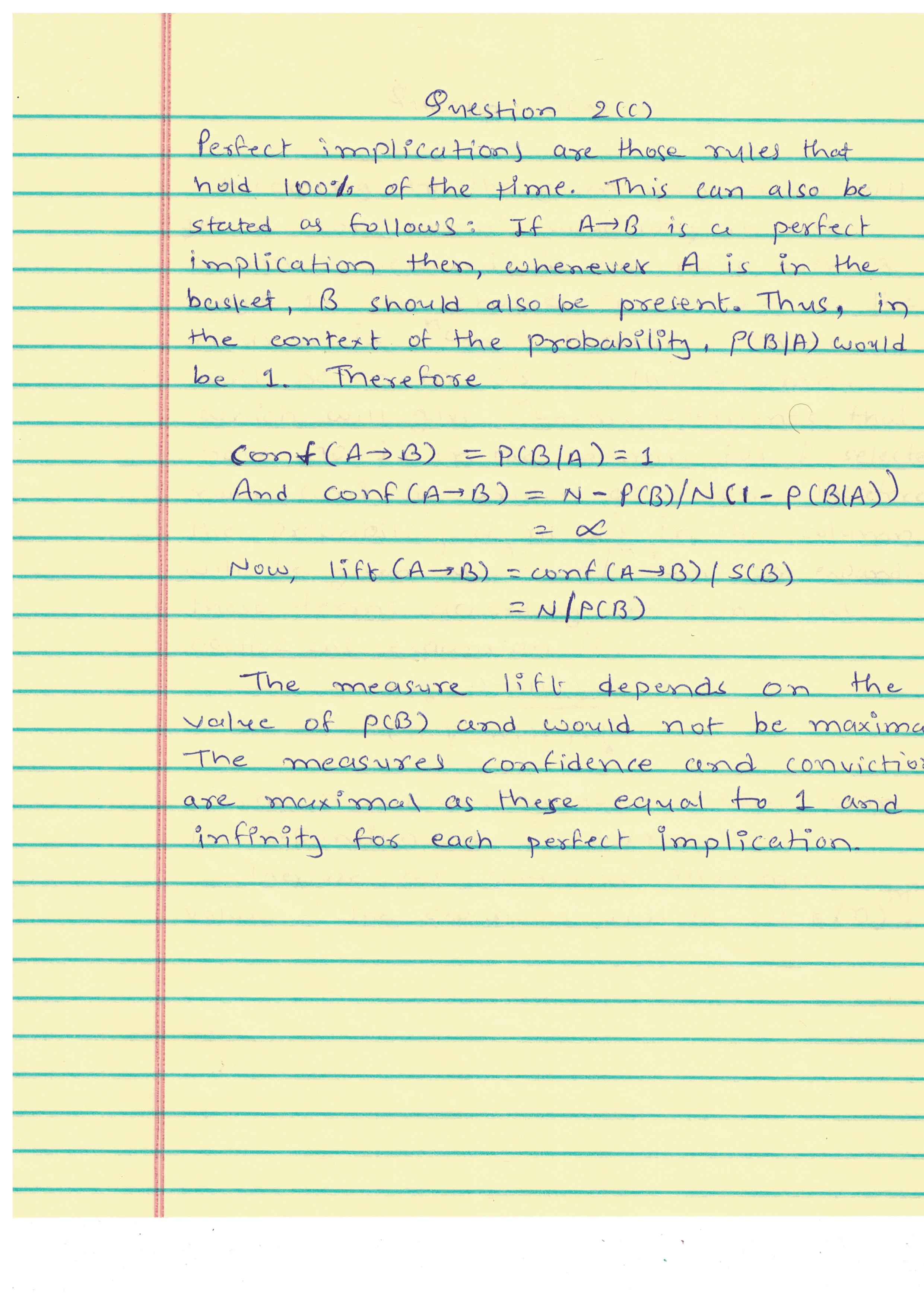
1. Question1.py is attached.
2. First of all we need to count the number of mutual friends that each pair of user have in such a network. For this, we need a map reduce job that functions like the map-reduce job for finding the frequency of words in a file.   
     
   Now, let us define our method such that it can look at the two names in the key to find equal keys. Therefore, the reducer gets as input a key denoting a pair of friends along with the list of their number of common friends. Then filter out pairs who were already friends directly. Then transform a new list by share count. Then aggregate the recommendation list by mutual friend count. Sort each users friend recommendation list by shared friend count in descending order and print and write to the file.

|  |  |
| --- | --- |
| 924 | [439, 2409, 6995, 11860, 15416, 43748, 45881] |
| 8941 | [8938,8943,8944,8945,8940] |
| 8942 | [8939,8946,8940,8943,8944] |
| 9019 | [320,9017,9021,9022,317,9023] |
| 9020 | [9021, 9016, 9017, 9022, 317, 9023] |
| 9021 | [9020,320,9016,9017,9019,9022,317,9023] |
| 9022 | [9019,9020,9021,317,9016,9017,9023] |
| 9990 | [9988,35667,9991,9992,13134,13478,13877,34299,34485,34642] |
| 9992 | [9987,9989,9988,9990,35667,9991] |
| 9993 | [9987,9991,35667,13134,13478,13877,34299,34485,34642,37941] |

**Answer to Questions 2(a)**



**Answer to Questions 2(b)**

**Answer to Questions 2(c)**

**Answer to Questions 2(d)**

Top 10 pairs by confidence:

The first column is the confidence score, the second column is the X in X ! Y , and the third column is the Y in X ! Y.

1.0 ["DAI93865", "FRO40251"]

0.999176276771005 ["GRO85051", "FRO40251"]

0.9906542056074766 ["GRO38636", "FRO40251"]

0.9905660377358491 ["ELE12951", "FRO40251"]

0.9867256637168141 ["DAI88079", "FRO40251"]

0.983510011778563 ["FRO92469", "FRO40251"]

0.972972972972973 ["DAI43868", "SNA82528"]

0.9545454545454546 ["DAI23334", "DAI62779"]

0.7326649958228906 ["ELE92920", "DAI62779"]

0.717948717948718 ["DAI53152", "FRO40251"]

**Answer to Questions 2(e)**

Top 10 pairs by confidence:

The first column is the confidence score, the second column is the X in X; Y ! Z, the third column is the Y in X; Y ! Z and the fourth column is the Z in X; Y ! Z.

1.0 ["DAI23334", "ELE92920", "DAI62779"]

1.0 ["DAI31081", "GRO85051", "FRO40251"]

1.0 ["DAI55911", "GRO85051", "FRO40251"]

1.0 ["DAI62779", "DAI88079", "FRO40251"]

1.0 ["DAI75645", "GRO85051", "FRO40251"]

1.0 ["ELE17451", "GRO85051", "FRO40251"]

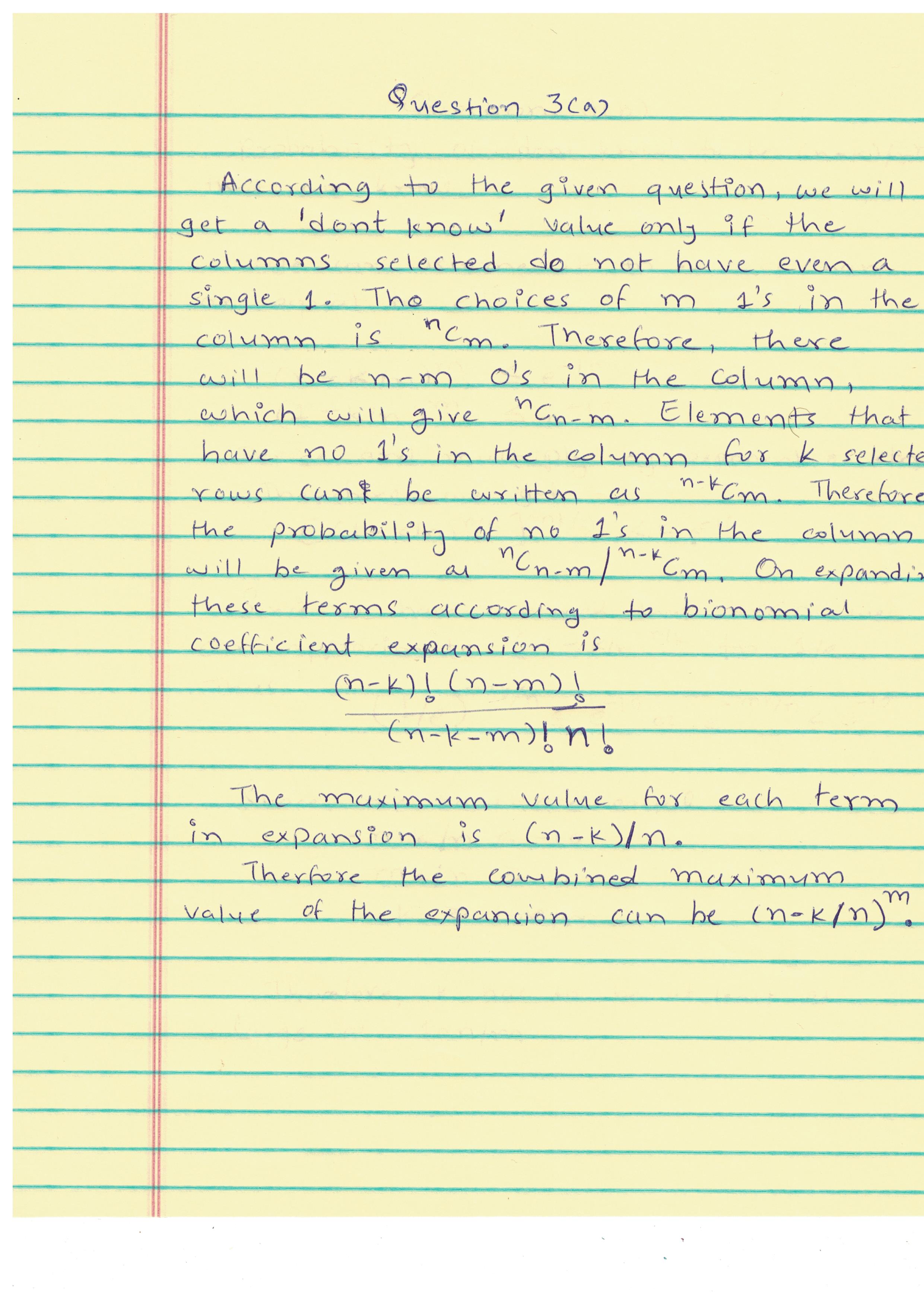
1.0 ["ELE20847", "FRO92469", "FRO40251"]

1.0 ["ELE20847", "GRO85051", "FRO40251"]

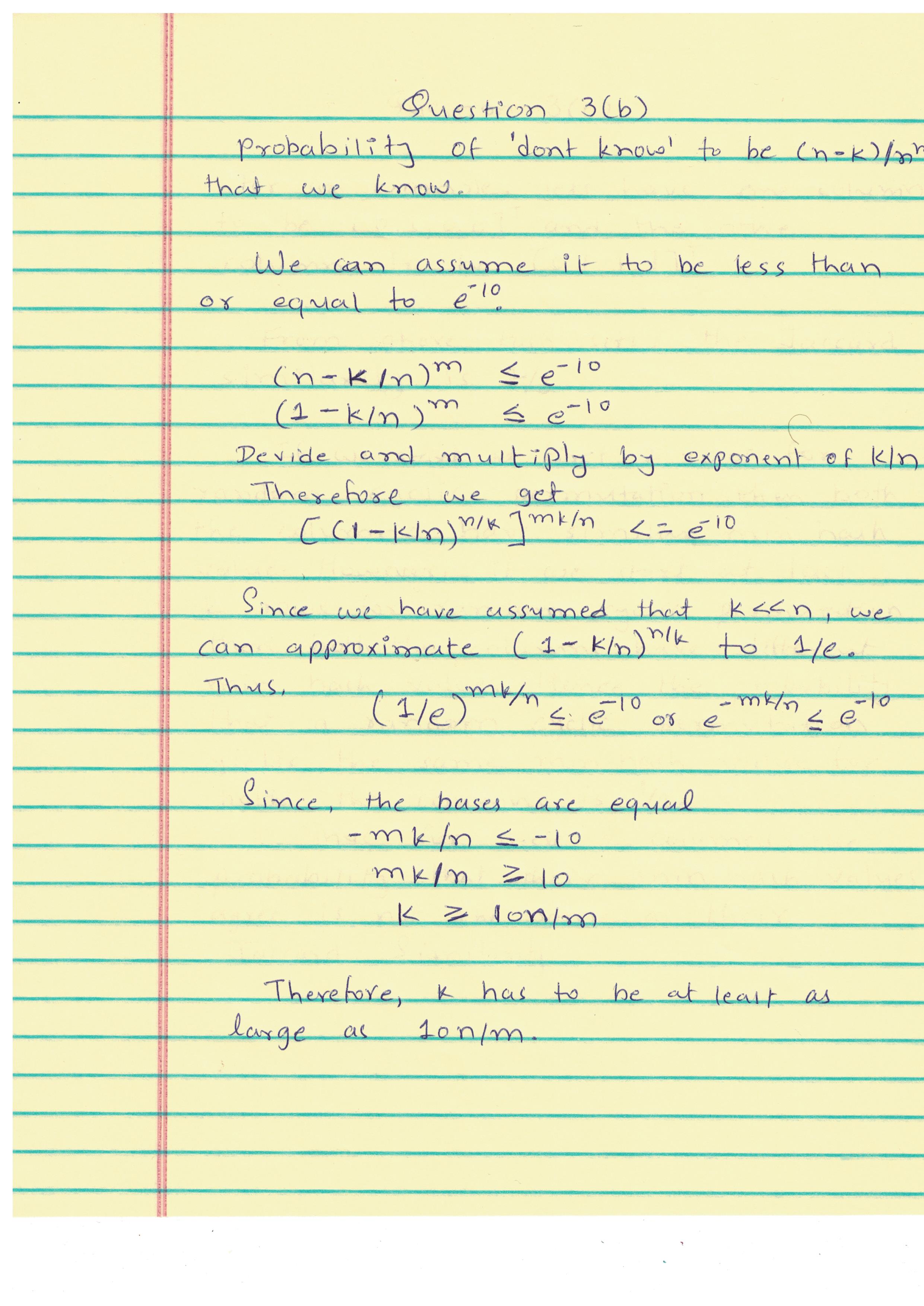
1.0 ["ELE26917", "GRO85051", "FRO40251"]

1.0 ["FRO53271", "GRO85051", "FRO40251"]

**Answer to Questions 3(a)**



**Answer to Questions 3(b)**



**Answer to Questions 3(c)**