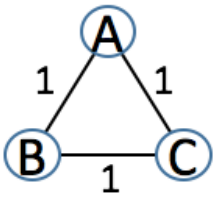
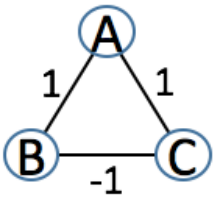
Assignment 05 – Graphs

For this assignment you will write a program to process graph data, and answer some questions about your results*.*

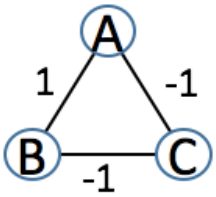
In chapter 5 of the text, we considered positive and negative relationships in networks, with triads of friends indicated by triangles of nodes. The diagrams to the right are similar, but in this case the edges represent trust. As with friendships, there are four possible relationships in a trust triad, represented by diagrams (a) through (d).



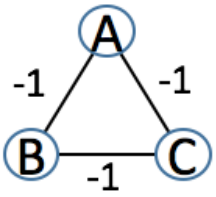
(a) A, B, C all trust each other



(b) A trusts B and C, but they don't trust each other



(c) A and B trust each other, but neither trust C



(d) None of them trust each other

For this assignment, you will study data from epinions.com that represents pairs of reviewers (the nodes) and whether they trust each other or not. Reviewers were asked whether they trusted or distrusted another reviewer. Each entry in the dataset has two numeric identifiers, representing the reviewer and reviewee, plus a 1 or -1; where 1 indicates the reviewer trusts the reviewee and -1 indicates that the reviewer distrusts the reviewee. The data can be used to create a graph that represents an undirected, signed network. The graph will have no duplicate or reciprocal pairs of nodes.

ASSIGNMENT:
Write a program to process the input file and identify triads. Your program should successfully run on file epinions1.csv within 15 minutes. (It should likely run in less than a minute).
*Note: Depending on the software package and/or algorithm you use, you may not be able to process the **epinions2.csv** file in a reasonable amount of time. (For example, when I used Python and networkx to retrieve all cliques and look at the ones of size 3 - the triangles - the program would take hours.) So, a smaller file, **epinions1.csv** is also provided, which has fewer than 60,000 entries. You should try the larger **epinions2.csv** file, but if that doesn't work, make note of that in your report and run your program on **epinions1.csv** instead. The small file **epinions0.csv** is the file used in the example.

Your program must follow the programming guidelines and the specifications provided below.

Also, write a one-page report documenting how you approached the problem, any problems you faced and any insights you gained. The report must include the following ***labeled*** sections:

PURPOSE:
INPUT:
OUTPUT:
WHAT THE PROGRAM DOES:
RESULTS: Describe and discuss the results of your analysis. You will notice that the actual distribution of triad types differs from the expected distribution based on random assignment. Discuss briefly how they differ and why that might be so.
ANY ADDITIONAL INFORMATION. (Anything you want to share -- students often talk about special issues that came up and how they addressed them, for instance. You may not have anything you want to include, and that is fine.)

The report should be submitted in .pdf or Word format, in a document ***labeled with your last name***, eg: Dugas_HW5_Report.pdf.

Zip your report, code, and a screen shot of ***the output of the runs of all files you successfully processed*** into a zip (compressed) file and submit in Canvas. ***Your screenshot for the epinions1.csv run must show run start and end time, and/or run duration.*** Your zip file must have your last name inside and out. I.E. it should unzip to a folder that also has your last name. *Do not submit the epinions csv files with your assignment. Screen shots may be included in your report, but also must be included separately in the zip file.*

PROGRAMMING GUIDELINES:

Programs will be screened for plagiarism. If you "borrow" code, be sure to document the details of the source; otherwise it will be considered plagiarism and result in a zero grade for the assignment. Borrowed code will not count toward your grade, only original code will be considered.

Programs should employ good programming practices. An example is the use of descriptive variable and function names.

Annotation and Comments: ***IMPORTANT***

- Program header must include **your** name and assignment information (use comments).
- Comments must also be used at the beginning of the program to give an overall description of the purpose of the program.
- Comments must also include detailed running instructions to run in a terminal window.
- Comments should also be used throughout the code to explain what it is doing. It should be possible to re-create your program based on the comments alone. Poorly commented programs will receive poor grades.

PROGRAM SPECIFICATIONS:

Do all processing using one program only.

All programming must be in python 3 unless otherwise arranged with the instructor.

If you use external python packages, please note that in your program comments in the run instructions.

Your program should be named: **lastname.py**

Your program should prompt the user to input a filename. This can be epinions0.csv, epinions1.csv, or epinions2.csv. Do not use an argument list.

As you process the input file, count the number of positive and negative reviews. These counts will be used to create an expected distribution of the four triad types to compare to the actual distribution. In creating the expected distribution of triad types, assume that the positive and negative trust values are randomly assigned. [See example below]

Identify all of the triads in the graph. Do not just count them: You will need to know the value of the edges in the triangle formed by the three nodes in the triad. For each triad, identify which of the four triad types it represents, and add to the appropriate count.

Output should be presented in a user-friendly format similar to the example shown. Do NOT list all the triads. Your output should contain the following:

1. Number of edges used to identify triads
2. Number of positive (trust) edges
3. Number of negative (distrust) edges
4. Probability p that an edge will be positive: (number of positive edges) / (total edges)
5. Probability that an edge will be negative: $1 - p$
6. Number of triangles
7. Expected distribution of triad types (based on p and $1 - p$ applied to the number of triangles in the graph). Show number and percent.
    a. Trust-Trust-Trust
    b. Trust-Trust-Distrust
    c. Trust- Distrust -Distrust
    d. Distrust- Distrust- Distrust
    e. Total
8. Actual distribution of triad types. Show number and percent.
    a. Trust-Trust-Trust
    b. Trust-Trust-Distrust
    c. Trust- Distrust -Distrust
    d. Distrust- Distrust- Distrust
    e. Total

------- EXAMPLE ------

This example was run on a much smaller file, epinions0.csv, which is provided on the Canvas site. The file contains 96 entries. You might use it as a practice file, to see that you get the same results, before running your program on the larger files.

*** START ***

RESULTS FOR FILE: epinions0.csv

triangles 27
TTT: 20 Edges used: 96

TTD: 3 trust edges: 68 probability %: 70.83
TDD: 3 distrust edges: 28 probability %: 29.17
DDD: 1 total: 96

Expected Distribution*

	percent	number
TTT:	35.54	9.60
TTD:	43.90	11.85
TDD:	18.08	4.88
DDD:	2.48	0.67

Actual Distribution

	percent	number
TTT:	74.07	20
TTD:	11.11	3
TDD:	11.11	3
DDD:	3.70	1

*** END ***

*remember to consider all edge combinations. e.g. for a triad with edges a, b, c with value t or d each, a triad of type TTD can represent t, t, d or t, d, t or d, t, t. So percent is 3 x .71 x .71 x .29 = 43.9%

The actual distribution differs quite a bit from the expected distribution that assumes trust/distrust is randomly distributed. There may be a number of reasons for that. In the assignment you are asked to think about that and share your thoughts on the reasons for the differences.

Here are the triads that were found, sorted by type:
Note: Do not print the triads in your assignment.

TTT		5	20	1		5	50	1		20	50	1
TTT		5	20	1		5	52	1		20	52	1
TTT		20	50	1		20	25	1		50	25	1
TTT		20	52	1		20	57	1		52	57	1
TTT		20	79	1		20	25	1		79	25	1
TTT		20	79	1		20	35	1		79	35	1
TTT		20	79	1		20	57	1		79	57	1
TTT		20	25	1		20	35	1		25	35	1
TTT		20	25	1		20	57	1		25	57	1
TTT		20	35	1		20	57	1		35	57	1
TTT		79	25	1		79	35	1		25	35	1
TTT		79	25	1		79	57	1		25	57	1
TTT		79	25	1		79	39	1		25	39	1
TTT		79	35	1		79	57	1		35	57	1
TTT		25	35	1		25	57	1		35	57	1
TTT		25	57	1		25	21	1		57	21	1
TTT		88	86	1		88	87	1		86	87	1
TTT		88	86	1		88	89	1		86	89	1
TTT		88	87	1		88	89	1		87	89	1
TTT		86	87	1		86	89	1		87	89	1
TTD		5	20	1		5	79	−1		20	79	1
TTD		20	52	1		20	25	1		52	25	−1
TTD		52	25	−1		52	57	1		25	57	1
TDD		8	6	−1		8	7	1		6	7	−1
TDD		20	52	1		20	23	−1		52	23	−1
TDD		20	23	−1		20	25	1		23	25	−1
DDD		52	23	−1		52	25	−1		23	25	−1