ALGORITHMS, DATA AND SECURITY A.Y. 2022/23
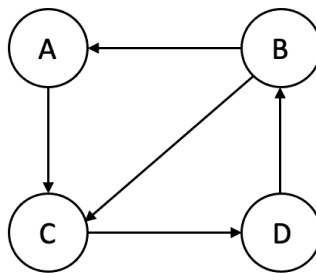## Final on May 10th, 2023

Surname _____   **Name** _____

ID number _____

Question 1 (points 7)
a) Describe what is a cluster centroid and how we can measure the distance between two clusters.
b) Explain the K-means clustering algorithm.
c) Discuss the pros and cons of the K-means clustering algorithm, also with respect to the hierarchical clustering algorithm.
d) How can we use this algorithm in a recommender system?

Question 2 (points 9)
a) Describe the PageRank algorithm and explain its basic update rule.
b) Apply the basic PageRank update rule for the first 2 iterations to the graph below, starting with page B having PageRank equal to 1 and all the other pages (A, C and D) with PageRank equal to 0.
c) When does the PageRank algorithm terminate? Considering the graph below, which node(s) do you expect to achieve the highest score and why?
d) Describe the goal of the centrality measure, explain a centrality measure of your choice different from PageRank and discuss how these two centrality measures differ.

Question 3 (points 9)
a) Describe 2 cybersecurity attacks of your choice among those we examined.
b) Explain how public key cryptography works.
c) Discuss the pros and cons of public key cryptography.
d) How do we use public key cryptography in the digital signature process?
e) Explain what is a blockchain and how blocks are linked with one another.

Question 4 (points 7)
a) Describe the goal and the steps of the collaborative filtering algorithm.
b) Explain how do we measure similarity between users.
c) Given the ratings shown in the following table, predict Tom's rating for Joker by applying the collaborative filtering algorithm.

	La La Land	Frankestein Junior	Avatar	Spider-Man	Joker
Tom	3	4	4	5	**?**
Mary	5	5		3	4
John	2		5	5	3
Lucy		4	2	4	5

The similarity values are as follows:

1.00000000	-0.8528029	0.8164966	0.08111071
-0.85280287	1.0000000	-0.6237783	-0.12105003
0.81649658	-0.6237783	1.0000000	-0.49669963
0.08111071	-0.1210500	-0.4966996	1.00000000

d) How do we speed up the collaborative filtering algorithm when we have to deal with millions of users and movies?