

Technical Test

Time Limit: 48 hours

For all following code, you must use Python. Please focus on solving one problem and only move to the next one if you have a satisfactory solution: we'd rather see one good solution than three partial ones. Once finished, create a ZIP file for all solutions and attach it to an email. Please include explanations for each solution.

1) A population of U users receives N notifications at specific times during the day. You are given, to represent who receives a notification and when, a CSV containing a list of pairs (user, notification_time). The similarity between users is defined as "how close in time they tend to receive notifications". As an example, if user A and user B both receive lots of notifications around 12:00 and not much during the rest of the day, they have high mutual similarity. If user C receives most notifications spread over the evening and not much around 12:00 he has low similarity to both user A and B. Please explain how you could quantify $\text{similarity}(\text{user1}, \text{user2})$ mathematically and implement a program that, given the attached CSV, will find the two most similar users.

2) A population of U users receives N notifications a day. You are given the data in a simple CSV file (attached) containing a list of pairs (user, notifications_per_day). Please write a program that implements an algorithm to sort the users by "notifications in a day". However, let's say you have limited RAM, so you are not allowed to work on more than 3 users at a time. For the sake of parsing simplicity, you are allowed to load the whole CSV in RAM, but the sorting cannot work on more than 3 users at a time.

3) A population of U users receives N notifications a day. You are given the data in a simple CSV file (attached) containing a list of pairs (user, notifications_per_day). Please write a program that implements the optimal algorithm to find the user that receives the second highest number of notifications in a day.