Vritee Muni (A0264771X)
Tasneem Binte Jamaludeen (A0255684R)
Evangeline (A0255488M)
Julie Lo Smithgul (A0264787J)

1. Explain the purpose of the algorithm in your app. Clarify whether the chosen algorithm is the only existing algorithm that can perform the intended task. List alternative algorithms and explain your choice.

The algorithm we chose is the Gale-Shapley algorithm. The Gale-Shapley algorithm, also known as the Stable Marriage Problem algorithm, is used to match two sets of elements where two parties have preferences for one another. [1] A version of the Gale-Shapley Algorithm may help us immediately rank matches to an individual based on their preferences and create a proposer-receiver relationship.

It primarily serves to streamline and optimize user experience by efficiently processing data (ie. major, personality, year of study, modules taken, etc.) and providing profiles that could be potential matches for study buddies. It also aims to enhance functionality by recommending profiles based on user inputs and interactions in the app. This is done through predictive analytics, data sorting and machine learning that trains the algorithm to deliver outcomes that are personalized according to the user's interaction. In doing so, this ensures that the app remains dynamic, responsive and adaptive to the user's needs to allow for a more satisfying and productive user experience.

However, there are other alternative algorithms such as:

1. **Elo Rating Algorithm**: Often used in games like chess, the Elo Rating System works to associate points to individuals based on their past matches. [2]
2. **Automated Decision-making Algorithm**: "Algorithmic or automated decision systems use data and statistical analyses to classify people for the purpose of assessing their eligibility for a benefit or penalty".[3]

Nevertheless, given the simplicity and efficiency of the Gale-Shapley Algorithm, we will be modifying the Gale-Shapley algorithm to better meet the needs of our app.

2. Explain how this algorithm works - what are the inputs and outputs, what is the sequence of steps performed on this input and leading to the output (you can illustrate this with a flow chart)?

The Gale-Shapley algorithm works by matching an individual from one group to an individual of another, such that every person is in a 'stable pairing'— that is— in each set of matchings there are no two people who prefer to be with each other than their current matchings.

---

[1] https://medium.com/aiskunks/understanding-gale-shapley-stable-matching-algorithm-and-its-time-complexity-4b814ee2642

[2] https://mattmazzola.medium.com/understanding-the-elo-rating-system-264572c7a2b4

[3] https://www.brookings.edu/articles/fairness-in-algorithmic-decision-making/

A rudimentary pseudocode of this algorithm is shown below:

```
function stableMatching {
Initialize all group_one_user ∈ group_one_dataframe and
group_two_user ∈ group_two_dataframe to free
while ∃ free individual group_one_user who still has individual
group_two_user to match to {
group_two_user = first individual on group_one_user’s list to whom
group_one_user has not yet match with
if group_one_user is free
(group_one_user, group_two_user) are matched
else some pair (group_one_user', group_two_user) already exists
if group_two_user prefers group_one_user to group_one_user’,
group_one_user’ becomes free
(group_one_user, group_two_user) become engaged
else
(group_one_user’, group_two_user) remain engaged
}
}
```

Initialisation:

To obtain the initial data necessary for the algorithm in our study buddy app, we will open up selection periods over regular intervals during the academic semester, and users will be *randomly* sorted into two groups.

1) Suppose the two groups are labelled 1 and 2, each having *n* number of individuals, where individuals of group 1 are labelled {A, B, C, D…}, and group 2 individuals are labelled {a, b, c, d…}. Users are then shown other user profiles from the complementary group, and asked to rank them according to personal preference (availability, major, courses taken, study style etc.). Following this ranking process, both groups will be put into two separate data frames, such that the first data frame contains group 1, as well as each of their personal rankings, and the second data frame includes individuals of group 2 and their personal rankings. These two data frames will be our **input variables.**

Loop:

After the selection period closes, the algorithm will work to pair each user such that everyone is placed into the most optimal pairing possible while minimising partner dissatisfaction. For the purposes of explanation, we can assume there is a pair (**A, a**), with preference (1,4) where **a** is first on **A**’s rankings, and **A** is fourth on **a**’s rankings.

2) Individuals from group 1 who are not yet matched will automatically be matched their first choice picks from group 2. Here, **a** is automatically ranked with **A** resulting from first choice preference.
3) This pairing is tentative, and will then be accepted or rejected by the algorithm based on the following conditions:

a) **Accept**, if group 2 individual is not yet matched.
b) If group 2 individual already has a match, then:
    i) **Reject**, if this pairing is worse than group 2 individual's current pairing
    ii) **Accept**, if this pairing is better than group 2 individual's current pairing

Termination:
Once all users are matched from each group, the loop is terminated, and all pairings have reached an optimal equilibrium.

3. What is (are) the limitation(s) of this algorithm(s)?
   The Gale-Shapley algorithm assumes fixed preferences, and doesn't account for the possibility that these preferences might change overtime. Therefore, data trained under the algorithm is most likely to reflect fixed preferences of majors, study timings, timetables, etc. Fixed preferences mean that everybody has a clear, consistent, and unalterable ranking of their potential matches. For instance, it is likely that the algorithm will pair two students in the same major, however there is a possibility of students changing their majors and not wanting to be paired with someone in the same major. Therefore, modifications in the algorithm must be made to account for changing preferences, to overcome the simplistic assumption that preferences are static and unchanging. There are fortunately ways to overcome this limitation through periodic preference updates or dynamic reassignment where the algorithm could allow for the reassignment of partners and accommodate the changed preferences.

   Another limitation of the algorithm arises due to its inconsideration of unmatched elements. Traditionally, the Gale-Shapley algorithm was used to match equal sizes of men and women on both sides. Nevertheless, if our app must focus on a one-to-one pairing, we must account for the possibility of unmatched individuals who may have to wait to be matched after another student with their preferences signs up on the app. Thus, it would be ideal to allow for matchings to occur with more than one partner to avoid this.

   Overall, while the algorithm has limitations that make the functioning of the app too simplistic, there are opportunities to modify and/or create hybrid forms of algorithms to overcome these limitations.

   References:

   1. Instability in Stable Marriage Problem: Matching Unequally Numbered Men and Women - Gui-Yuan Shi et.al
   2. The Challenge of the Stable Marriage problem - University of California, Santa Barbara

You are expected to devolve into the technical details of how the algorithm works using what you have learned in class. The level of technical depth should be similar to that at which PageRank was covered in class.
Your report should take the form of a user-friendly documentation to ensure that non-group members can easily understand your project and your algorithm and promptly start collaborating with you.