**ASSESSMENT**

Python Full-stack Stream, assessment test 2 hours

|  |  |  |
| --- | --- | --- |
| **SECTION TYPE** | **TOTAL MARKS AVAILABLE** | **NOTES** |
| **Complexity Analysis** | 25 | Multiple questions, all comprising 25 total |
| **Algorithms 1 (Coding)** | 25 | 1 question only |
| **Algorithms 2 (Coding)** | 25 | 1 question only |
| **Concept Generation & Prototype (Design)** | 25 | 1 question only. Essay-based answer is needed |
| **100 marks available total** | | |

**Please note:**

This is a closed book exam - 2 hours in length.

You are allowed to use PyCharm or any other tool as you see fit (but no googling! Unless it's for looking-up documentation or syntaxes: other than that, please don’t outright cheat as we will know + it’s good to be ethical!).

You are given this PDF and / or Word document - you also have access to a **Python file** (e.g. ‘code environment’) that can be accessed through this [**link**](https://drive.google.com/file/d/1pDuwHwQgjXggXmYswuUxLZycTh3AC1c9/view?usp=sharing). Feel free to download a copy and make use of it. This **.py** contains skeleton code for you to modify, for the algorithm questions, as well as sample test cases that you can use to test your solution!

Please leave your code in the **.py** file, and your submission for Q1 and Q4 in **either .py file** (please just document / format it nicely for easier marking) or as a **separate marked file** (e.g. word document). You don’t need to use the **.py** file (since the real aim is just to receive your code), but it's still recommended regardless as the **.py** file has helpful test cases and is formatted for you to use.

Also - when submitting, make sure to rename all the files you’re passing to **include your name** - this makes marking a lot easier!

Finally - best of luck! Remember, this is not a determinant of your talent. This test is simply to assess your ability, see where you can grow, as well as push you out of your comfort zone. Stay calm, deep breaths, you got this!

**In case the hyperlink used for accessing the .py file doesn’t work, make use of the below!**

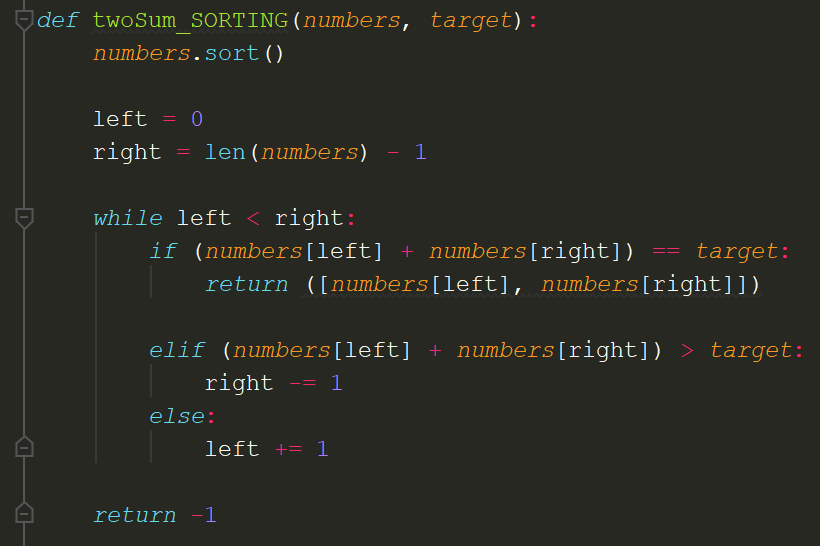
<https://drive.google.com/file/d/1pDuwHwQgjXggXmYswuUxLZycTh3AC1c9/view?usp=sharing>

*`Questions begin on the next page*

|  |  |
| --- | --- |
| **Complexity Analysis** | **25 total (each question is worth 5)** |

**Describe the average Big O complexity (both Time and Space) of the following code samples and / or problem samples (no code). Each answer is worth 5 marks (2.5 for Time, 2.5 for Space).**

**Code Sample 1 (5 marks):**

****

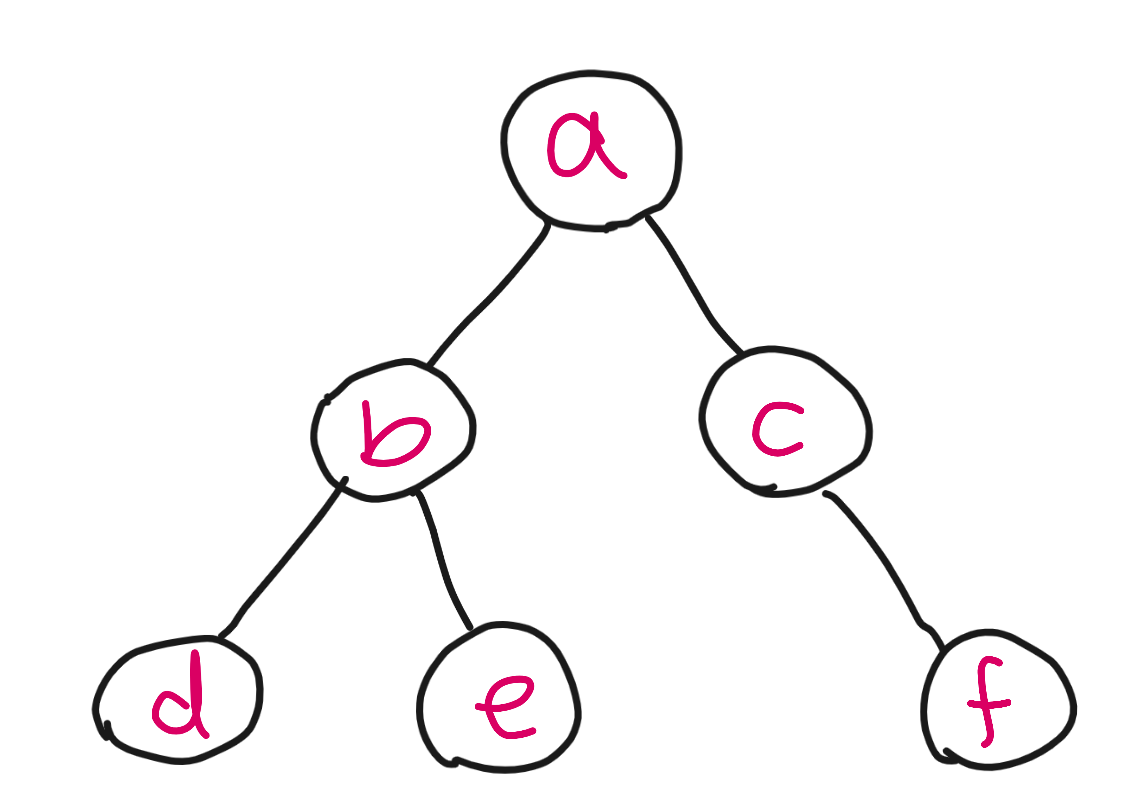
This is a sorting-based approach for solving the Two Sum problem. Given an input array of integers and a target, it scours the array until it’s able to find two numbers that add up to the target. If such a pair can be found, then it returns it as an array containing two elements. If no such pair can be found, then it returns an integer -1 to denote that no such pair exists.

What is the Big O Time and Space complexity of this code?

Time = O(n)

Space = O(1)

**Problem Sample 2 (5 marks):**

****

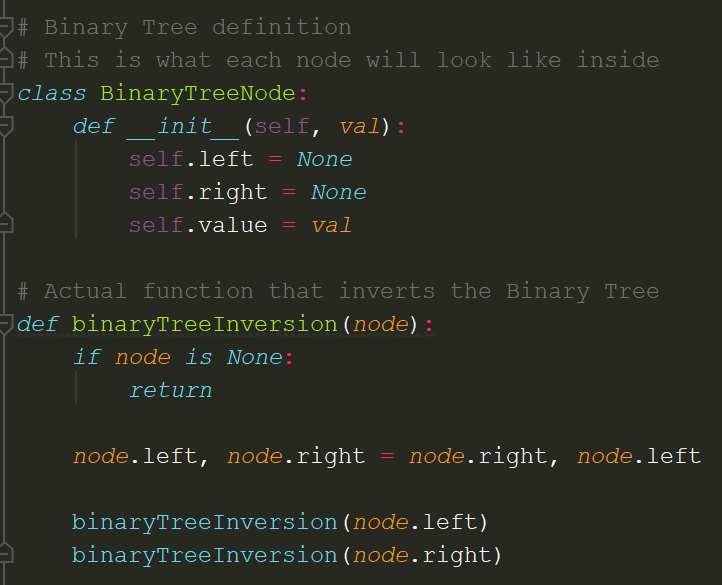
I have a Graph data structure that has N number of nodes (a small sample representative of it is depicted via the image above). My hypothetical code runs a Breadth First Search algorithm on it - that is, I’d like to traverse all nodes in order to store their values (e.g. add node.value to my collection). For my algorithm, I use a queue data structure to hold the order of the nodes (e.g. visit node A, then node B, then node C). My algorithm is effectively just BFS.

What is the Big O Time and Space complexity of this code?

Time = O(N)

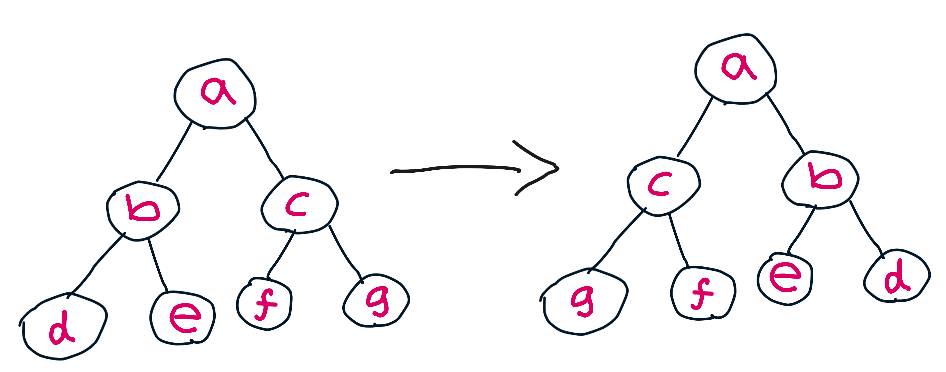
Space = O(N)

**Code Sample 3 (5 marks):**

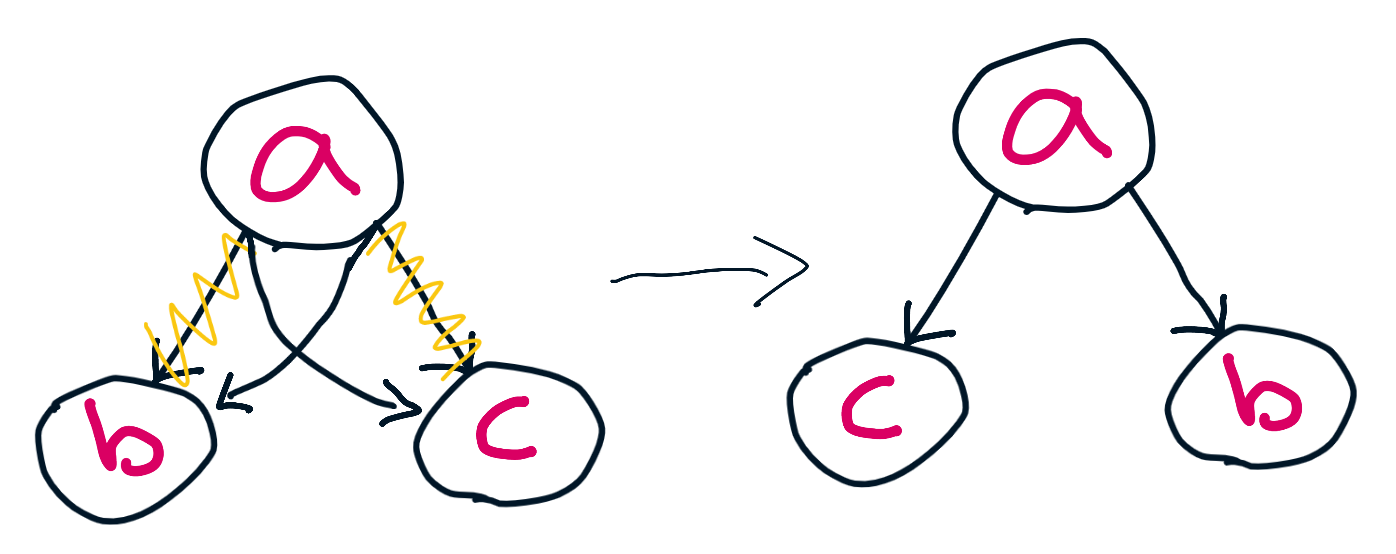
****

I receive a classic interview question, invert a binary tree, and have devised an algorithm that can correctly achieve this. This is accomplished through my **binaryTreeInversion** function (the definition above it, **binaryTreeNode**, is just to clarify what each node has in terms of attributes). The outcome of this code causes the position of all nodes to be flipped - nodes on the left side go on the right, with the right nodes going to the left.

See the example below for a visual representation:



My code operates by going to each node, and making it flip its pointer references around - essentially, it tells it that **node.left** should refer to nodes on the opposite side and vice versa. Essentially:



What is the Big O Time and Space complexity of my **‘binaryTreeInversion’** code?

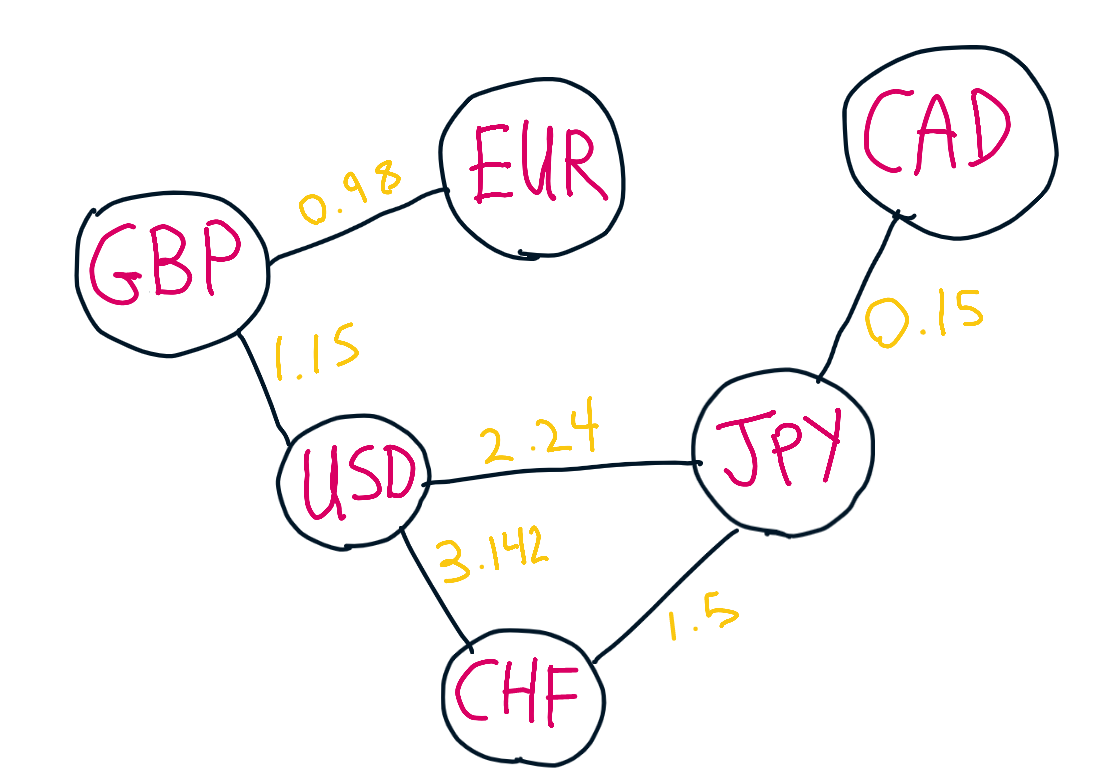
*Hint:* *Remember, the code is a recursive function! Consider how this can impact Big O!*

Time = O(n)

Space = O(n)

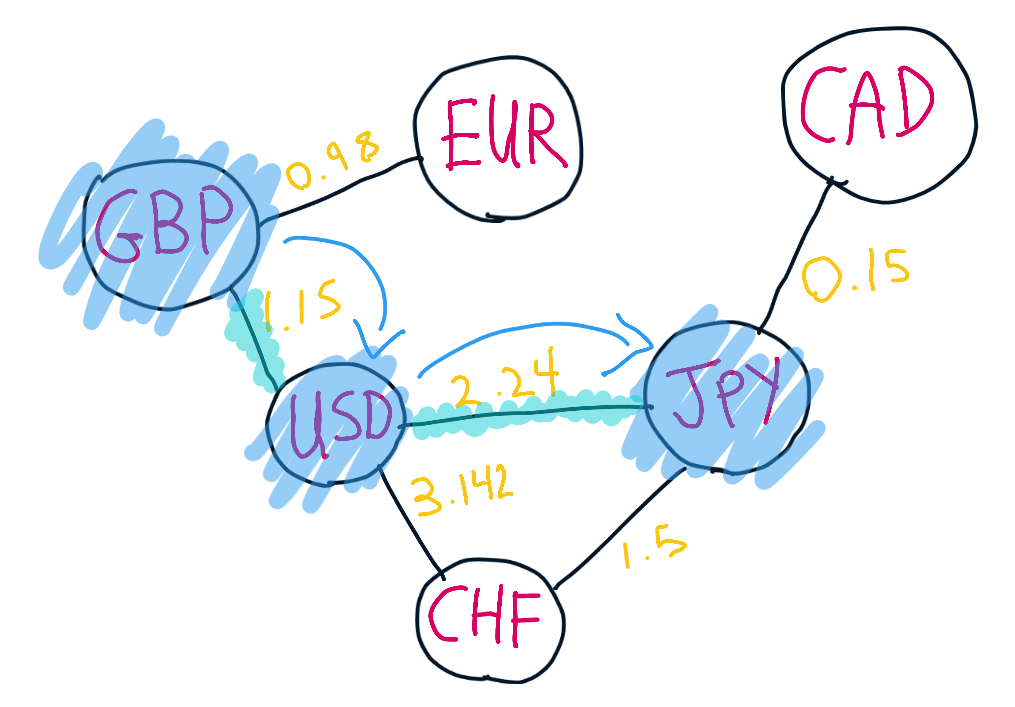
**Problem Sample 4 (5 marks):**

No image for this problem unfortunately. I am given a graph where each node represents **real-life currencies (e.g. GBP, USD, JPY)** - these nodes are connected to **some** other currencies (e.g. **GBP** may be connected to **USD**, but not **JPY**). The connections between each node represents a ‘currency conversion’ (e.g. changing from **GBP** to **USD** will cost me 1.14 conversion exchange rate based on its connection). See below for a visual example:



Here, you can see how the currencies are connected to each other - with each connection having a ‘exchange rate cost’.

I am asked to check how many ‘conversions’ it takes to change from one currency to another (through traversing many nodes - e.g. **GBP** to **USD**, **USD** to **JPY**). Effectively - an algorithm that checks for the shortest path between node A and B. So for currencies like **GBP** to **JPY,** the answer would be **2** (as I have to traverse through 2 connections!):



*Technically we can go* ***GBP -> USD -> CHF -> JPY*** *but we don’t as it’s unnecessary - why take a longer path?*

By the end, my algorithm is successful in finding the number of hops required. In the instances where there is no possible path between two given currencies, I simply return **-1** to denote that it’s not possible.

For my code, I use **BFS**; I start at one node, and breadth first search all adjacent nodes over and over again until I find the ‘destination’ node - whilst keeping a counter that tracks how many ‘hops’ I’ve made so far. I then output this counter as the representative of the shortest possible path.

Please **also note that the dataset is based on real-life currencies; there are no made-up currencies at all.** The entirety of the dataset **consists solely of all the currencies existing in the world right now.**

Given all of this, what is the Big O Time and Space complexity of this code?

Time = O(n2)

Space = O(n)

|  |  |
| --- | --- |
| **Algorithms 1 (Coding)** | **25 (1 question)** |

Given two strings **stringA** and **stringB**, return **true** if **stringA** **is an anagram of stringB** (else return **false**!). Note that an anagram is where the other word has all the original letters, except just in a different order.

For example, some sample input and outputs would be:

|  |  |  |  |
| --- | --- | --- | --- |
|  | **stringA value** | **stringB value** | **Output value** |
| **Sample Input 1** | code | odec | true |
| **Sample Input 2** | code | cardeo | false |

Please note:

* String inputs (**A** and **B**) will be alphanumeric and may contain both upper and lower cases (e.g. **abc** or **ABC** or **aBc**)
* For the sake of this question, **different case letters are treated differently. As in, ‘A’ will not be equivalent to ‘a’.** So **‘code’** and **‘odec’** are anagrams, but **‘cOde’** and **‘odec’ are not** as the uppercase **O** is not equivalent to its lowercase version.

*In your answer, please discuss your solution - what is its Big O Time & Space complexity? Why have you chosen this approach? Could there be a more efficient way (and if so, how)?*

*If you are short on time, you can also submit pseudocode or simply describe what solution you’d write in code (just describe what you have in your mind) - this cannot attain full marks, but it is still a perfectly acceptable answer and can get partial marks.*

*In essence, just submit what you have even if you don’t know the answer!*

|  |  |
| --- | --- |
| **Algorithms 2 (Coding)** | **25 (1 question)** |

Write a function that, given a **sequence** of elements, returns **true** if any value **appears twice or more** in the array sequence. If all elements are unique / distinct though, then the function should return **false**.

You can assume that the elements will be strictly **either** integers or strings only - an integer **1** will not be considered equivalent to string **“1”** though. You can

For example, some sample input and outputs would be:

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Sequence value** | **Output value** | **Why?** |
| **Sample Input 1** | [“a”, “b”, “c”, “a”] | false | Element “a” occurs twice - hence, the sequence is not fully distinct |
| **Sample Input 2** | [“a”, 1, 2, “b”, 3] | true | No element exists twice or more |

Please note:

* There are multiple solutions that can be provided for this question - due to its ease / low-level difficulty, the **brute-force approach can only obtain 15 marks maximum** (where its runtime is **O(N^2)** time).
* For full marks, try thinking of a solution that can achieve even better performance in terms of time (even if it potentially comes at the cost of space).

*In your answer, please discuss your solution - what is its Big O Time & Space complexity? Why have you chosen this approach? Could there be a more efficient way (and if so, how)?*

*If you are short on time, you can also submit pseudocode or simply describe what solution you’d write in code (just describe what you have in your mind) - this cannot attain full marks, but it is still a perfectly acceptable answer and can get partial marks.*

*In essence, just submit what you have even if you don’t know the answer!*

|  |  |
| --- | --- |
| **Concept Generation & Prototype (Design)** | **25 (1 question)** |

You have been tasked with creating a social networking mobile app (e.g. Instagram, BeReal, etc). When I (as a user) go onto this application, I should be able to interact with my friends by seeing what’s going on in their lives (e.g. a moment of their life through a Instagram story snapshot / BeReal post / some sort of post).

You will not need to code anything, only design the application. As this is a vague and fairly open-ended question, consider the following points to include in your ‘design’ answer:

* Consider - who is the user? Do they have a persona? Who are you catering for?
* Design the application - what platform (web, mobile)? What may it look like (simple sketch is fine, or word-description)? What colours would you use and why? Think of this design like the way you approached the ‘Design the perfect door’ task for your Theory Assignment
* What design heuristics and principles will you be following? How does your design meet or fail to meet them?
* What made you choose this design?
* What are some future considerations you may need to make for your design? You can google for this question areas like “Future trends of UX / UI” (**only this area - please).**

Users of a social networking app are most likely to be in the demographic of teens to 30s, as older generations are more likely to use not have smartphones or prefer methods such as texting familiar with technology while younger people tend not to be as interested. However, as younger people often use social media by lying about their age, I would make a version for children with safety filters, which must be managed by a parent’s account. It would have the same interface as the adult one, so children feel they are getting the same experience.

People like to use social media on the go, for example when commuting, and also at home, so the application should have both mobile app and web versions. It should also be installable onto computers, so users don’t have to visit the website every time they want to use the app, reducing clutter and risk of accidentally closing the page. It should also follow similar layouts on both mobile and computer for ease to use. Users should be able to create accounts so they have their own profile which is useable on any device and saves their content. Income of users can range, so there should be a basic free version that’s useable so people try the app in the first place, and paid options which offer additional benefits to encourage users to upgrade if they can afford it. Paid options could offer benefits such as no advertisements. All versions should have the ability to search for and add friends, message other users, take and share content such as images and videos, share music and video links from the internet, and search for and view others content, with users being able to save, like, and comment on content as well as share it with other users. There should be a personalised dashboard showing content from account’s that the user follows. The app should notify users via email when someone has requested to add them. Autogenerated content based on what the user has interacted with should also be suggested, but settings should allow the user to stop this and only see posts from people they are friends with if they want. To encourage people to use it as opposed to other apps, it should have unique features, such as more customisation/personalisation options than other forms of social media. To expand on this concept, I would make my app unique by making it syncable with music playing from other apps such as Spotify or Apple Music, as well as having it’s own integrated music player as part of the paid option, so artist’s get paid for having their music on the app and are more likely to allow it. As different colours have different moods, instead of having a consistent background, I would use a background colour that matches the song playing, as people can experience a variety of emotions while listening to music and songs also have different moods. I would do this by matching the colour to the cover art of the single/album playing, as cover art colours are likely to be consistent with the mood of the song. For example, if a slow, peaceful song is playing, the artwork for it may be an image of the ocean. Therefore, the background would be a similar shade of blue so the colour of the application matches the song and therefore the mood. Contrast is also important, so depending on the background colour, buttons and text would either be black or white depending on which has greater contrast so that it stands out and doesn’t clash. When there is no song playing, for example when the user first logs in, I would make it so the user could choose a default between dark or light mode, with blacks/dark greys or whites/light greys so the interface isn’t too distracting and can be standardised with other applications. In settings, the user would be able to customise their default background colour.

It would follow Norman’s heuristics by having a simple, easy to use interface with mapping and visible, standardised icons such as pause/play music, message, and search for friends. It would be designed for error, for example if a post is accidentally sent prematurely it would be easy and intuitive to delete it, with a cross icon in the corner. It would follow F and Z pattern’s by having important functions such as a search bar and the ability to go back at the top, other users posts in the centre, and the details about the song such as length, artist name, song name, and functions such as pause/play, shuffle, skip, and repeat at the bottom. For users in countries with right-to-left language, it should all be inverted as a default because their eyes are likely to naturally scan the other way, but whatever the country it should be possible within settings to change language and/or invert the page as someone may naturally look right-to-left but live in a left-to-right language country or have a different first language for example. It would follow Nielsen’s principles by having recognisable icons, e.g making status visible by having a green dot by the names of online friends, or showing whether a song is paused with the standard double line icon. It would also use familiar language such as “like” and “post”. This also makes the app consistent to industry standards. It should have efficiency, for example by being able to quickly scroll through posts or find a friend faster through suggestions when typing in the search bar, based off of friends of friends or search history. The design should be minimalist to prevent an overload of information and allow the user to quickly be able to perform a function without having to find where to do it through clutter. It should be clear to understand why an error has occurred and how to fix it, for example if something won’t post due to the wifi having lost connection an error message and explanation should appear telling the user to reconnect.

The future of UX/UI design includes things like VR and AR, so if the app is already allowing people to take and post photos, they should be able to add filters, see things with AR, and in the future view holograms. Voice control such as Siri is already popular, so it’s also important that the app is compatible and the user can type posts with text-to-speech. There should also be an element of personalisation, so as well as being able to choose between dark and light mode and background colours, there could be future options such as animated backgrounds. Gesture recognition may also become popular in the future, so for example instead of having to scroll with their hands, a user could use their gaze. For accessibility as well as preference, things like font size, font, language, and contrast should be customisable.