Task 1

You are a developer who accidently went back in time. You can’t return back to your timeline because you Time machine need to sort array from 10 Gigabytes of unsorted integer data. The problem in front of you is Machine during that time was not that advanced. You find a device with 1 gigs of ram and 240 Gigabyte of hard drive.

-They are all integers [0,65,535] like 10,5,98,32,22,87,5,6,9 etc.

-Same integer values can be repeatedly appearing in the file.

Create a function to read 10 gigabytes of data and sort it. (Pseudo code and logic will also work)

Task 2

We are a small startup company which offers the precise and concise news and weather information to our users. We are not news or weather agency that collects data. We filter the responses from various sources and show it on our mobile app. We need a backend application which fetches data from various sources and sends to our user’s phone.

Although we provide weather informations without any authentication but we need authentication to access news api.
So we implement /signup, /login , *logout functionality. You are free to manage user session however you want*
For registration we need user’s email, password and name.

Endpoints

News(/news) : This api is can only be accessed with authenticated (login users). The Api fetch news or top headlines from newsapi (https://nwsapi.orge/). Also implement search using keyword functionality in case no keyword fetch the top headlines and in case of query fetch query related news(`/news?search=bitcoin`).

```
{
“count”:10,
“data”:[
      {
    "headline": "Human organs can be stored for three times as long in major breakthrough for transplants",

    "link": "https://www.telegraph.co.uk/science/2019/09/09/human-organs-can-stored-three-times-long-major-breakthrough/"

  },
{
    "headline": "IRS goes after cryptocurrency owners for unpaid taxes",

    "link": "https://www.cbsnews.com/news/own-bitcoin-irs-pursues-cryptocurrency-owners-for-unpaid-taxes/",

  }.

   ...

]

}
```

Weather(/weather) :This api doesn’t need authentication and is needed to fetch weather forecast over next five days. Use Openweathermap API to fetch weather forecast data. The response must contain following data, (you can choose to hard-code the location).

```
{

“count”:5,

“unit”: ”metric”,

“location”: “Banglore”,

“data”:[

      {

         "date": “Sun March 06 2020”,
         "main": “Rain”
         "temp": 293.55,


      },

      {
         "date": “Sun March 07 2020”,
         "main": “Sunny”
         "temp": 294.64,


      },.

      .......

]

}
```

Constraints

There are some constraints that you should be aware of. Not completing any of the following constraints will stop your candidacy from moving forward:

- You must use **JavaScript (Node 10 +)** .
- You must return the specified fields that are in the sample requests above in JSON format. You can add more fields if you'd like.
- This needs to be a running JavaScript application on your localhost that serves an HTTP request not a console application.
- Write core unit tests for these endpoints using (Jest, Supertest or Mocha & Chai)
- Share documentation to API in Swagger / Postman
- You must write documentation for project setup in markdown.

Assessment

Primarily, we will be assessing good **design decisions**. To do this, we have another third party api and we want to implement with your authentication model and other functionality.

Secondly, we will be assessing good **application performance**. The overall speed of the application should not be affected by poor API speed. You need to be creative on how to solve this problem. (Hint: Consider caching the responses from third party api)

In addition, we will be assessing the following points:

- JavaScript Proficiency, Clean code principles
- Lint usage ES / TS Lint
- Ability to understand and use 3rd party APIs
- Ability to parse different forms of data
- Ability to write unit tests
- Ability to write documentation
- Generally professional code that follows standards in matter of commits and security

Submission

You should upload your code to a Github repository (private or public) and share it with the Tech Alchemy team (arjun@techalchemy.co, soham@techalchemy.co and saksham@techalchemy.co). Your repository should have a README.md that explains how to run the code and if you’ve done anything extra. **If you fail to produce this repository within the time period mentioned in the email you received your application will be rejected.**