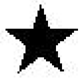
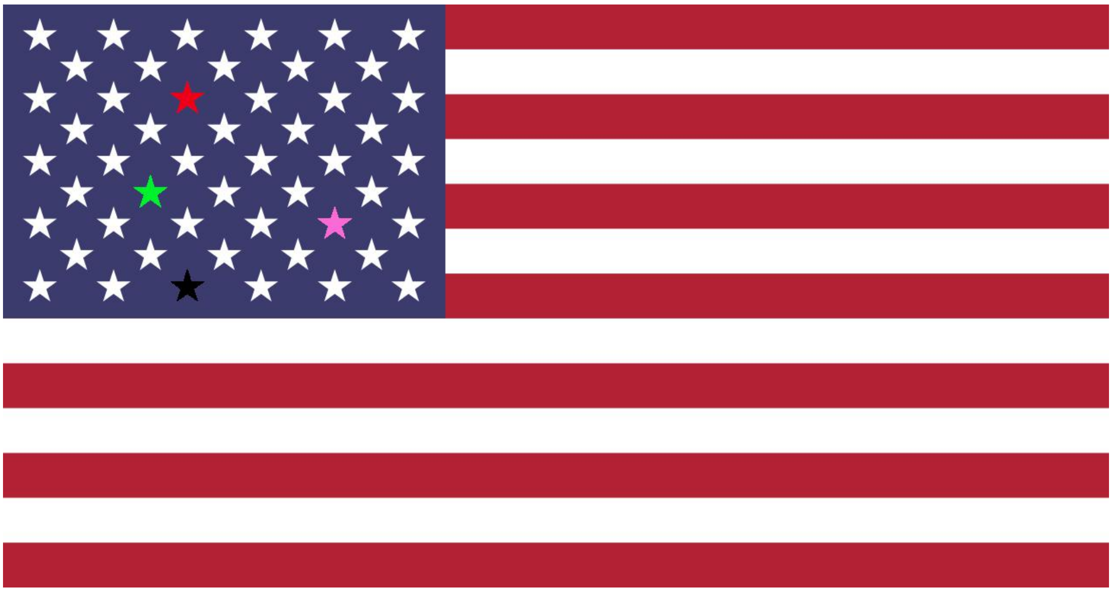
SYSTEMATRIX

Data Scientist Interview Question Packet

In what follows, you are asked to complete a set of both programming and conceptual questions. For the programming section you have two options: You may either complete (**Question 1 <u>AND</u> Question 2)** **<u>OR</u> Question 3** only. For the conceptual section, you must complete **Question 4 <u>AND</u> Question 5**. Good luck!

1. Computer Vision [**Programming**]: Use "mask.png" (the black star below) to find all the stars (regardless of color) in "us_flag_color.png" and draw bounding boxes around each star. Please include the code you used to solve this problem in your response. Ensure that your code is written in Python, R, or Java.





2. [Programming - Python]: You have a function that takes a Pandas DataFrame (*db*) and a list of lists (*row_id_pairs*) as input, then returns a DataFrame (*data*), where each row of *data* is composed of two rows drawn from *db* based on their row_id. Pairs of *row_id*s are specified within *row_id_pairs* as follows: [[1,54], [6,83], ...]. Rows within *db* have the following schema: *row_id, var_1, var_2, ..., block_id*. The *block_id* must be dropped from the left-hand row before forming row pairs. The block_id should remain on the right-hand row.

```
def create_entity_pairs(db, row_id_pairs):
        data = pd.DataFrame()
        for pair in row_id_pairs:
                row_left = db[db['row_id'] == pair[0]].reset_index(drop=True)
                if isinstance(row_id_pairs, list):
                        row_left = row_left.drop('block_id', axis=1)
                row_right = db[db['row_id'] == pair[1]].reset_index(drop=True)
                row = pd.concat([row_left, row_right], axis=1)
                data = data.append(row)
return data
```

Improve the above function by making it faster. Please include the data set you used to benchmark the comparison between the two functions and the time difference between them. You can use any data set you would like, so long as it adheres to the schema above and exceeds at least 10000 rows.

3. Create a branch out Force-Directed Graph [Programming]:

Using d3.js or a JavaScript visualization library of your choice, create an interactive force directed graph of character co-appearance in Les Misérables. However, instead of displaying the entire graph at once, start by just showing one node at the beginning (say, the character Marguerite) . Upon user action on that node (such as a double-click), the graph showed be updated to show her immediate neighbors (Fantine, Valjean). Clicking on Fantine now should show her own neighbors, and so on. Make sure to keep track of the nodes and links already in the network, so there are no duplicate nodes created on a click action. Further, any links between nodes just added to nodes already in the network must be shown. The end result of completely exhausted graph should have the same structure (nodes and links) as the graph here. Along with your code, please provide a brief write-up of your algorithm. Feel free to use any color scheme and aesthetics for the graph!

4. Information Retrieval: Imagine you are asked to collect data on all the businesses in the European Union (~23 million) that you can, but are restricted to only what you can obtain via scraping/crawling. The data you have been tasked to retrieve includes Business Name, Address, E-mail, Phone, and Social Media Profiles. How would you go about obtaining these data? Do you foresee any potential issues/pitfalls in this approach? If so, how would you address them? Please lay out the step-by-step process you would take to retrieve these data, calling out specific techniques, data sources, and pseudocode, if applicable. No programming is required for your answer.

5. Natural Language Processing: Imagine you have been given millions of rows of address data and are asked to clean these data. While reviewing the data, you notice that the fields of the addresses are often poorly separated. The below examples show the three major problems, where the red string is problematic and the green string is the corrected version.

Problem #1 (Concatenation):

John Smith, 12 Main, StAtlanta, GA30303

John Smith, 12 Main, St, Atlanta, GA, 30303

Problem #2 (Improper Element Separation):

McDonalds 526 P, once De Leon A, ve Atlanta, GA, 30308

McDonalds, 526 Ponce De Leon Ave, Atlanta, GA, 30308

Problem #3 (Improper Field Separation)

The Wren’s Nest 1050 Ralph David, Abernathy Blvd, Atlanta, GA 30310

The Wren’s Nest, 1050 Ralph David Abernathy Blvd, Atlanta, GA, 30310

Describe the strategy/algorithm you would employ to solve these three problems. Please be as specific as possible.