

Back-end Engineer

TECHNICAL TEST

Part 1: Coding

Candidate can choose the language they're familiar with and should come up with their best solution in term of algorithm complexity.

1. Given an integer n , we can construct a matrix of $n \times n$ numbers from 1 to n^2 .

For example with $n = 3$ we have this matrix: 1 2 3 4 5 6 7 8 9

Your task is to write a function, receive N as input, and print (to stdout) all the numbers of the matrix in the spiral pattern, starting from 1. Your code should use as less memory as possible.

Example:

Input: $n = 3$. Output: 1 2 3 6 9 8 7 4 5

Input: $n = 4$. Output: 1 2 3 4 8 12 16 15 14 13 9 5 6 7 11 10

1. Design a data structure that holds a list of integers that support 4 methods
 - `push_back(n: int)`: insert an element to the end of the list
 - `push_front(n: int)`: insert an element to the beginning of the list
 - `pop_back()` → int: get and remove the last element
 - `pop_front()` → int: get and remove the first element

All methods should have $O(1)$ time complexity, you should not use any predefined data structure in your programming language.

Example:

Initially, the list is empty `list = []`

`push_back(8)`, `list = [8]`

`push_back(15)`, `list = [8, 15]`

```
push_front(20), list = [20, 8, 15]
push_back(30), list = [20, 8, 15, 30]
pop_back(), list = [20, 8, 15] and return 30
pop_front(), list = [8, 15] and return 20
```

3. Given an array A of N integer. For each position i ($0 \leq i < N$), you have to print (to stdout) the number P_i = product of all element except A[i]. Your code should use as less memory as possible.

You can assume the product of all elements and all P_i is less than 2^{60}

Example

A = [2, 1, 3, 4, 5]

Output: 60 120 40 30 24

4. Given a 2D array A of N*N integers. If an element in this array equal 1, set its entire row and column to 1 and return the modified array. Your code should use as less memory as possible.

Example

```
A = 0 1 2 3
    3 1 2 4
    1 0 2 3
    5 9 2 5
```

Return:

```
A = 1 1 1 1
    1 1 1 1
    1 1 1 1
    1 1 2 5
```

Part 2: System design

Design the architecture of the following system. Your solution should contain traffic estimation, tools/technologies to use, services and databases design, bottleneck discussion. You can make any assumptions to drive your solution. Choose one of the decisions in your design and explain why do you go with it rather than other alternative options.

1. Design a marketplace like [Chotot.vn](https://chotot.vn).
2. Design a ride-hailing app like Uber, Grab.
3. Design a chat app like Zalo.
4. Design a translator app like Google translate.
5. Design an image recognition app (using camera to recognize the objects - table, dog, cat, human, etc).

Part 3: Real-world application

Implement the following web application and deploy to some hosting environment (e.g. Heroku). Submit the source code and URL to the application. You're free to add more backend or frontend features for bonus points. Your application should be written from scratch and use any languages/tools of your choice. The front-end can be very simple.

1. A web app to record the current IP and location of the browser. The app should display current visit's information (IP address, location, timezone...), the last 100 visitors, and also the top 100 IPs that visit the website the most.
2. A secret note keeper that allows users to login via one social media app (Google, Facebook, Twitter, etc) and then keeps their notes privately.
3. A simple text game that rolls a random number each time you play. You should enter your name before playing. There should also be a leaderboard feature to display the people with the highest scores.
4. A web app that displays a new random famous quote every day (UTC timezone) and allows people to like/unlike the quote. The app should display the current like/unlike count.
5. A web app that displays one image, and allows anyone to replace the image with another image. The app should also show the links to the 5 recent past images.