

## QUESTIONS

### Normal

1. Describe Git branching strategies (Git-flow, single branch, feature branch etc.) which you have used and what purpose does it serves.

**ANS** - Single branch เป็น branch สำหรับผู้เริ่มต้นใช้งานที่เข้าใจง่าย เพราะมีแค่ branch เดียว

- Feature branch เป็น branch ที่มีการแยกออกเป็น main และ feature branch ซึ่งจะมีการ merge กับ main ในภายหลัง
- Git-flow คล้ายกับ feature branch แต่มีจำนวน branch มากกว่าและซับซ้อนกว่า
- Forking เป็น branch ที่แยกออกมาจาก single branch การแก้ไขใดๆ จะไม่เกี่ยวข้องกับ branch เดิม

2. How do you revert a commit that has already been pushed and made public?

**ANS** ใช้คำสั่ง git revert โดย undo commit ที่ถูก push ไปก่อนหน้านี้แล้วสร้าง commit ใหม่

3. How do you normally solve conflicts in a feature branch before merge?

**ANS** ใช้คำสั่ง git rebase เพื่อให้ code ใน feature branch base จาก master อันล่าสุด

4. “200 OK” what does it mean and show use case this HTTP Status?

**ANS** server ที่ทำการ request ไป response กลับมาว่าสำเร็จ ตัวอย่าง ถ้าใช้ HTTP method แบบ GET ไปที่ server server จะ response กลับมาเป็นข้อมูลใน server นั้น

5. “201 Created” what does it mean and show use case this HTTP Status?

**ANS** server response ว่าได้ทำการเพิ่ม resource สำเร็จแล้ว มักเป็น response จาก HTTP method แบบ POST ตัวอย่าง เพิ่มข้อมูลไปที่ server sever จะ response กลับมาโดยแสดงข้อมูลที่เพิ่งเพิ่มเข้าไปพร้อมกับข้อมูลเดิม

6. “301 Moved Permanently” what does it mean and show use case this HTTP Status?

**ANS** server ที่ทำการ request ไป response กลับมาว่า location ได้เปลี่ยนไปแล้ว ตัวอย่าง ใช้ HTTP method แบบ GET ไปที่ server ด้วย location A server จะ response กลับมาด้วย HTTP status 301 แล้วบอก Location B พร้อมข้อมูลใน location นี้

7. “400 Bad Request” what does it mean and how to identify the problem?

**ANS** server ที่ทำการ request ไป response กลับมาว่าทาง client ทำการ request ไม่สำเร็จ วิธีตรวจสอบให้ลองดูคำสั่งที่ request ไปว่าพิมพ์ถูกต้องหรือไม่ หรือติดเรื่อง permission หรือไม่

8. “401 Unauthorized” what does it mean and how to identify the problem?

**ANS** server ที่ทำการ request ไป response กลับมาว่าทาง client ต้องการ permission สำหรับ request นี้ วิธีตรวจสอบให้ลองทำการ login user สำหรับ permission นี้ก่อนส่ง request

9. “403 Forbidden” what does it mean and how to identify the problem?

**ANS** server ที่ทำการ request ไป response กลับมาว่าทาง client ไม่มี permission สำหรับ request นี้ วิธีตรวจสอบให้ลองดู permission ของ user นี้ว่าสามารถ request ได้หรือไม่

10. “404 Not Found” what does it mean and how to identify the problem?

ANS server ที่ทำการ request ไป response กลับมาว่าไม่สามารถหา resource เช่น webpage เจอตามที่ request ได้ วิธีตรวจสอบให้ลองดู URL ที่ request ถูกต้องหรือยังใช้งานได้อยู่หรือไม่

11. “500 Internal Server Error” what does it mean and how to identify the problem?

ANS server ที่ทำการ request ไป response กลับมาว่าไม่สามารถทำตามที่ request ได้เนื่องจากปัญหาทาง server วิธีตรวจสอบให้ลอง request ในภายหลังหรือ refresh เพื่อ clear cache ก่อน request อีกครั้ง

12. “502 Bad Gateway” what does it mean and how to identify the problem?

ANS server ที่ทำการ request ไป response กลับมาว่าไม่สามารถทำตามที่ request ได้เนื่องจากปัญหาทาง communication ของ server ผ่าน gateway วิธีตรวจสอบให้ลอง request ในภายหลังหรือ refresh เพื่อ clear cache ก่อน request อีกครั้ง

13. “503 Service Unavailable” what does it mean and how to identify the problem?

ANS server ที่ทำการ request ไป response กลับมาว่าไม่สามารถทำตามที่ request ได้เนื่องจาก server ไม่สามารถใช้งานได้ขณะนี้ วิธีตรวจสอบให้ลอง request ในภายหลังหรือลองหาข้อมูลดูว่า server นี้กำลังทำการ maintenance อยู่หรือไม่

14. “504 Gateway Timeout” what does it mean and how to identify the problem?

ANS server ที่ทำการ request ไป response กลับมาว่าไม่สามารถทำตามที่ request ได้เนื่องจากปัญหาทาง communication ของ server ผ่าน gateway ที่เมื่อไม่ได้รับ response หลังจาก request ไปสักระยะหนึ่ง วิธีตรวจสอบให้ลอง request ในภายหลังหรือ refresh เพื่อ clear cache ก่อน request อีกครั้ง

15. What are Linux network tools do you use for troubleshooting network problems as well as usage scenarios for each tool?

ANS – ping tool ตัวนี้ใช้สำหรับการตรวจสอบการเชื่อมต่อ network โดยการส่ง request แล้วดู response time  
– ifconfig tool ตัวนี้ใช้สำหรับแสดงข้อมูลของ network interface  
– netstat tool ตัวนี้ใช้สำหรับแสดง network ที่กำลังใช้งานอยู่

## Intermediate

1. Assume we have an application that is designed as below. Our application stopped responding due to an extremely high number of clients in some circumstances.
  - We have tried scaling a number of API Gateway and Service A nodes but it didn't help. What are the possible problems that lies in our system in which components and how to fix them?

ANS ใช้ Cluster Autoscaler ใน Kubernetes เมื่อ Pod มาอยู่ pending state เนื่องจาก resources ไม่พอ จะทำการตรวจสอบ nodes ในระบบก่อน ถ้าไม่มี nodes ให้ลง จะส่ง request ไปหา cloud ที่ใช้ในระบบนี้เพื่อขอ nodes เพิ่ม

2. How do you keep the docker image smallest as possible?

ANS เลือกใช้ base image ที่มีขนาดเล็ก เช่น ตัวที่ลงท้ายด้วย alpine และ build แบบ multi-stage (อาจใส่คำสั่ง rm -rf บางคำสั่งที่ไม่ต้องการ ใน docker file เพื่อช่วยลดขนาด image เล็กน้อย)

3. How does the Kubernetes service talk to each other in the same cluster?

**ANS** ถ้าเป็น Pod to Pod ใช้ overlay network โดยมี service เป็นตัวกลาง ซึ่ง Pod จะคุยกับ service ก่อนผ่าน kube-proxy ก่อนไปคุยกับอีก Pod

4. What's different between L2, L4, and L7 Load balancers? When to use it?

**ANS** - L2 หรือ Data link layer ใช้กับการรับส่งข้อมูลผ่าน MAC address (physical layer)

- L4 หรือ Transport layer ใช้กับการรับส่งข้อมูลผ่าน IP address

- L7 หรือ Application layer ใช้กับการรับส่งข้อมูลที่ซับซ้อนผ่าน DNS, HTTP (Domain name)

## Professional

1. Assume that you are using a private cloud for your infrastructure. How do you manage logs, metrics, and alerts for your infrastructure and applications? Which tools do you use and why?

**ANS** เลือกใช้ Grafana loki สำหรับการจัดการ logs เพราะใช้พื้นที่ในการเก็บ logs น้อยกว่า เนื่องจากเก็บแค่ metadata ไม่ใช่ full text และสามารถส่ง resulting alert ไปยัง Prometheus Alertmanager ได้

เลือกใช้ Prometheus สำหรับการเก็บ metrics แบบ time series ตัว Prometheus สามารถ Integrate กับตัวอื่นๆ อย่าง Grafana mimic เพื่อแสดง data visualization หรือส่ง alert ไปที่ Alert manager ได้

เลือกใช้ Prometheus Alertmanager สำหรับ alert เพราะสามารถ integrate กับ Grafana Loki และ Prometheus ได้

2. How do you secure the following?

- application
- infrastructure
- data

3. Based on your experience, how do you reduce your service downtime as much as possible during

- software upgrade
- database migration
- incident

4. How do you design your Kubernetes cluster? what DNS, CNI, ingress is being used? Why?

5. How do you design your Cloud infrastructure? (Answer as draw.io or your tool of choices)

- Networking
- Security

- Data Durability
  - High Availability
  - Autoscaling
6. How do you measure service quality to give the best experience to your customer? (SLO, SLA)
7. How do you design a highly scalable Kubernetes cluster and which Kubernetes objects are being used? (answer as drawio or your tool of choices)

### Hands-On

1. Please build and deploy your application of choice using Kubernetes. It should support multiple environments (e.g. beta, prod).

The solution must include the following

- Architecture Diagram
- IAC Scripts (CDK.TF or Pulumi) to provision an infrastructure
- Dockerfile or Buildpacks supports the following languages:
  - .Net Core
  - Go
  - Java
  - Node
  - Rust

CI/CD pipeline file:

- You can choose your CI/CD tool:
  - Gitlab CI
  - Github Actions
  - CircleCI
  - Jenkins
  - Travis CI
  - GoCD
  - BitBucket
  - Azure DevOps
- Kubernetes Manifests (Deployments, Services, Config Maps, Secret Management (Secrets or Cloud Managed Vault))
- Kubernetes Ingress Configuration

Please update README for steps on how to test, build, deploy and run the application using Kubernetes (answer as draw.io or your tool of choice with source code and demo)

## Algorithms

Choose your favorite language (Rust would be an advantage) to solve the following challenges:

1. Given an input string `s` and a pattern `p`, implement regular expression matching with support for `.` and `*` where:

- `.` Matches any single character.
- `*` Matches zero or more of the preceding element.

The matching should cover the entire input string (not partial).

### Example 1:

Input: `s = "aa", p = "a"`

Output: false

Explanation: "a" does not match the entire string "aa".

### Example 2:

Input: `s = "aa", p = "a*"`

Output: true

Explanation: `*` means zero or more of the preceding element, 'a'. Therefore, by repeating 'a' once, it becomes "aa".

### Example 3:

Input: `s = "ab", p = ".*"`

Output: true

Explanation: `.*` means "zero or more (`*`) of any character (`.`)".

Constraints:

- $1 \leq s.length \leq 20$

- $1 \leq p.length \leq 30$
- **s** contains only lowercase English letters.
- **p** contains only lowercase English letters, '.', and '\*'.
- It is guaranteed for each appearance of the character '\*', there will be a previous valid character to match.

**ANS** Code สำหรับข้อนี้เป็นไปตามด้านล่าง

```
def matching(s, p):
    if s.islower() and (p.islower() or p == ".*" or p[0] == "." or p[1] == "*"): #ตรวจสอบ input ว่ารับ
        ตัวอักษรพิมพ์เล็กเข้ามาหรือไม่แล้วสร้าง array dp ที่มีค่า false ทุกตำแหน่งยกเว้นตำแหน่งที่ row = 0 และ column = 0
        dp = [[False] * (len(p) + 1) for _ in range(len(s) + 1)]
        dp[0][0] = True
        #ถ้ามีการรับ input '*' เข้ามาที่ตำแหน่งที่ 2 ของ p จะให้ค่าที่ตำแหน่ง row ที่ 0 และ column ที่ 2 = True
        for i in range(1, len(p) + 1):
            if p[i - 1] == '*':
                dp[0][i] = dp[0][i - 2]
            #เช็ค input รับเข้ามาว่าที่ตำแหน่งตัวแรกและตัวที่สองของทั้ง input s และ p เหมือนกันหรือไม่ หรือ input p ได้รับมาเป็น
            ".*"
            for i in range(1, len(s) + 1):
                for j in range(1, len(p) + 1):
                    if p[j - 1] != '*':
                        dp[i][j] = dp[i - 1][j - 1] and (s[i - 1] == p[j - 1] or p[j - 1] == '.')
                    else:
                        dp[i][j] = dp[i][j - 2] or (dp[i - 1][j] and (s[i - 1] == p[j - 1]) or p[j - 1] == '*')
        print("Output: ", dp[len(s)][len(p)])
    else:
        print("inputs are incorrect")
#ทดสอบฟังก์ชันโดยคำสั่งด้านล่าง
s = str(input("Input: S = ", ))
p = str(input("Input: P = ", ))
matching(s, p)
```

2. Bob has a server farm crunching numbers. He has nodes servers in his farm. His company has a lot of work to do. The work comes as a number workload which indicates how many jobs there are. Bob wants his servers to get an equal number of jobs each. If that is impossible, he wants the first servers to receive more jobs. He also wants the jobs sorted, so that the first server receives the first jobs. The way this works, Bob wants an array indicating which jobs are going to which servers.

Can you help him distribute all this work as evenly as possible onto his servers?

### Example

Bob has 2 servers and 4 jobs. The first server should receive job 0 and 1 while the second should receive 2 and 3.

```
distribute(2, 4) # => [[0, 1], [2, 3]]
```

On a different occasion Bob has 3 servers and 3 jobs. Each should get just one.

```
distribute(3, 3) # => [[0], [1], [2]]
```

A couple of days go by and Bob sees a spike in jobs. Now there are 10, but he hasn't got more than 4 servers available. He boots all of them. This time the first and second should get a job more than the third and fourth.

```
distribute(4, 10) # => [[0, 1, 2], [3, 4, 5], [6, 7], [8, 9]]
```

**ANS Code** สำหรับข้อนี้เป็นไปตามด้านล่าง

```
def distribute(total_servers, total_jobs):
    #หาจำนวน job ที่ต้องใส่ต่อ 1 server
    jobs_per_server = total_jobs // total_servers

    #หาจำนวน job ที่เกินมาเมื่อแบ่งให้ server ไม่ลงตัว
    extra_jobs = total_jobs % total_servers
    #กำหนด array เปล่า
    actual_jobs_per_server = []

    #ตรวจสอบว่าแต่ละ server จะมี job เท่าไร
```

```

temp = extra_jobs
for i in range(total_servers):
    if temp > 0: #ถ้ามี job เกินมาให้แบ่ง Job ที่เกินใส่ไปที่ server ไ่จากตัวแรกสุด
        job_qty = jobs_per_server + 1
        temp -= 1
    else: #ถ้าไม่มี job เกินมาให้แบ่งลงทุก server เท่าๆกัน
        job_qty = jobs_per_server
    actual_jobs_per_server.append(job_qty)
print("Actual jobs per server are ", actual_jobs_per_server)

```

```

#กำหนด array เปล่า
distribution = []

```

```

# เอา job ลง server ตามลำดับ
step = 0
for i in range(total_servers):
    server = []
    for j in range(step,(actual_jobs_per_server[i]+step)):
        server.append(j)
    distribution.append(server)
    step += actual_jobs_per_server[i]
print("Distribution for jobs to each server is ",distribution)

```

#ทดสอบฟังก์ชันโดยการใส่คำสั่งด้านล่าง

```

distribute(2,3)
distribute(3,3)
distribute(4,10)

```



3. It's tricky keeping track of who is owed what when spending money in a group. Write a function to balance the books.

- The function should take one parameter: a dict with two or more name-value pairs which represent the members of the group and the amount spent by each.
- The function should return a dict with the same names, showing how much money the members should pay or receive.

**Further points:**

- The values should be positive numbers if the person should receive money from the group, negative numbers if they owe money to the group.
- If value is a decimal, round to two decimal places.

**Example:**

3 friends go out together: A spends \$20, B spends \$15, and C spends \$10. The function should return an object/dict showing that A should receive \$5, B should receive \$0, and C should pay \$5.

```
var group = {  
  A: 20,  
  B: 15,  
  C: 10  
}
```

```
splitTheBill(group) // returns {A: 5, B: 0, C: -5}
```

**ANS** Code สำหรับข้อนี้เป็นไปตามด้านล่าง

```
def splitTheBill(group):
```

```
    total_cost = sum(group.values()) #รวมค่าใช้จ่ายทั้งหมด
```

```
    avg_cost = total_cost / len(group) #หาค่าใช้จ่ายเฉลี่ย
```

```
    #นำค่าใช้จ่ายของแต่ละคนมาหักกับค่าใช้จ่ายเฉลี่ยแล้วแจกแจงของแต่ละคนพร้อมแสดงค่าด้วยทศนิยม 2
```

```
    ตำแหน่ง
```

```
    result = {name: round(cost - avg_cost, 2) for name, cost in group.items()}
```

```
    return result
```

```
#ตรวจสอบโดยการใส่คำสั่ง
```

```
group = {"A": 20, "B": 15, "C": 10}
```

```
result = splitTheBill(group)
```

```
print(result)
```

#### 4. Fibonacci

The Fibonacci numbers are the numbers in the following integer sequence.

0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, .....

In mathematical terms, the sequence  $F_n$  of Fibonacci numbers is defined by the recurrence relation

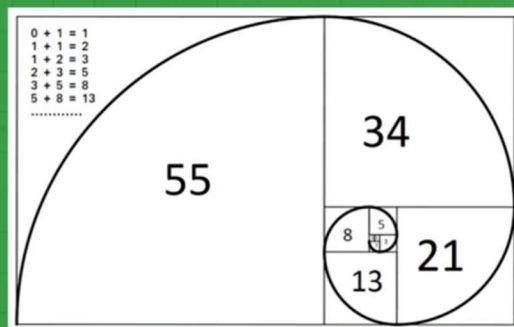
$$F_n = F_{n-1} + F_{n-2}$$

with seed values

$$F_0 = 0 \text{ and } F_1 = 1.$$

Given a number  $n$ , print  $n$ -th Fibonacci Number.

## Program for Fibonacci numbers



## Examples:

Input : n = 2

Output : 1

Input : n = 9

Output : 34

[ANS](#) Code สำหรับข้อนี้เป็นไปตามด้านล่าง

```
def fibonacci(n):  
    if n == 0: #ถ้า F ที่ n=0 จะได้ค่า 0  
        return 0  
    elif n == 1: #ถ้า F ที่ n=1 จะได้ค่า 1  
        return 1  
    else:  
        return fibonacci(n-1) + fibonacci(n-2) #ถ้า F ที่ n=1 จะได้ค่า 1  
n = int(input("Input: ")) #รับค่าเป็นจำนวนเต็ม  
if n > 0: #ถ้าเป็นจำนวนเต็มบวก ให้ใช้ฟังก์ชันคำนวณค่า Fibonacci และแสดงค่า  
    output = fibonacci(n)  
    print("Output:", output)  
else: #ถ้าไม่ใช่จำนวนเต็มบวกให้บอกว่าไม่ใช่ แล้วไม่คำนวณ  
    print("Input is not correct number")
```

## 5. Palindrome

Given a string, write a function to check if it is palindrome or not.

- **In case: even** you can reverse the string
- **In case: odd** you should use the middle to split the word and then reverse the string

A string is said to be palindrome if the reverse of the string is the same as the string. For example, “abba” is Palindrome, but “abbc” is not Palindrome.

Anna, civic, kayak, level, madam, mom, noon, racecar, radar, redder, refer, repaper, rotator, 12321, 15651

**ANS** Code สำหรับข้อนี้เป็นไปตามด้านล่าง

```
def palindrome(text):
    if len(text) <= 1: #ถ้าความยาวของ text มีแค่ 1 ตัวอักษรหรือไม่มี จะส่งค่า True ออก
        return True
    elif len(text) > 1: #ถ้าความยาวของ text มีมากกว่า 1 ตัวอักษร จะทำคำสั่งข้างล่างต่อ
#ตรวจสอบตัวอักษรตัวแรกและตัวสุดท้ายว่าตรงกันหรือไม่ ถ้าตรงกันก็ตรวจสอบตัวที่สองและตัวรองสุดท้ายต่อ ทำไปจนจบที่ตัวกลางของ
text
#ระหว่างนี้ถ้าเจอตัวที่ไม่ตรงกันจะส่งค่า False ออก แต่ถ้าตรวจสอบจนครบแล้วจะส่งค่า True ออก
        for i in range(len(text) // 2 + 1):
            if text[i] != text[-i + 1]:
                return False
        return True
#คำสั่งด้านล่างเป็นการทดสอบว่า text ที่ใส่ไปเป็น Palindrome หรือไม่
text = input("Enter text: ")
if palindrome(text):
    print(text, "is a palindrome")
else:
    print(text, "is not a palindrome")
```

## **How to Submit this Technical Home Test (in your Github Repository)**

### **Output format on Github**

jumpbox-part-time or jumpbox-contract or jumpbox-intern

|-> README.md

|-> question

|-> hand-on

|-> h01

|-> h02

|-> h03

|-> algorithms

|-> a01

|-> a02