



Every problem
has a solution.
You just
have to be creative
enough to find it.

Travis Kalanick, Uber co-founder



**¿Te gustaría ser contratado
por una importante empresa
de Tecnología?**

Reto Semestral

**en la Unidad de Formación TC2038 -Análisis y Diseño
de Algoritmos Avanzados**

Semestre Ago-Dic 2021

¿Qué buscan las empresas de Tecnología para contratarte?



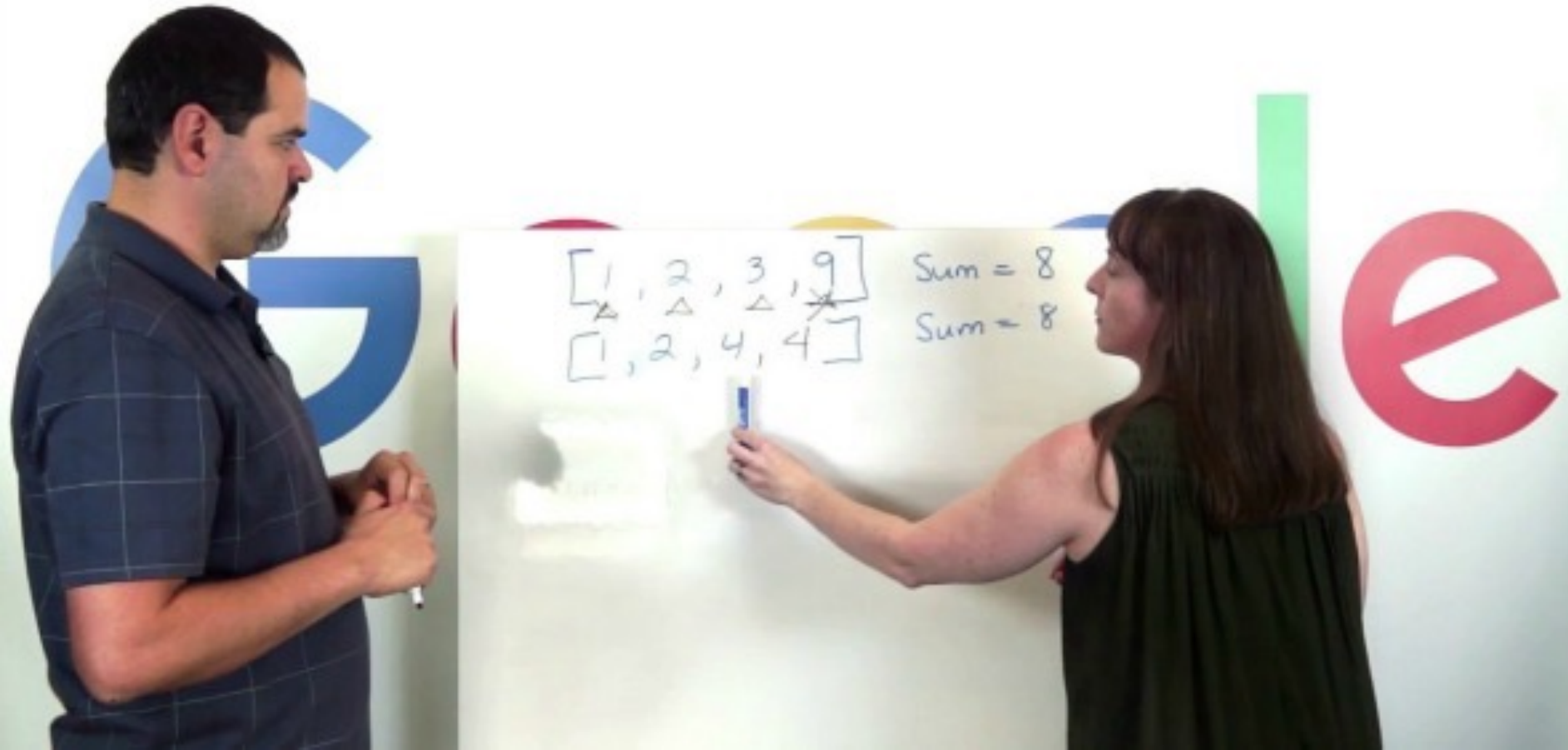
Entre otras cosas, tiene un peso importante tu **habilidad para resolver problemas con programación...**

Esto implica:

- Saber **programar** y hacerlo **eficientemente**
- Dominar uno o varios **lenguajes de programación**
- Diseñar y utilizar eficientemente **estructuras de datos**
- Aplicar **técnicas de diseño de algoritmos**



**... y en las entrevistas de trabajo,
normalmente te ponen a prueba
con el reto de resolver un pequeño
problema y programar su solución...**



**En nuestro curso,
incrementaremos las habilidades
para enfrentar este tipo
de entrevistas de trabajo...**



- **Ya sabes programar (tienes un pensamiento algorítmico)...**
 - En el curso fortaleceremos el hacerlo más eficientemente.
- **Ya conoces y utilizas varios lenguaje de programación...**
 - En el curso ampliaremos el nivel de dominio.
- **Ya conoces las estructuras de datos básicas y sabes diseñar nuevas...**
 - En el curso profundizaremos en su uso y ampliaremos sus aplicaciones.
- **...y aprenderás nuevas técnicas de diseño de algoritmos, teniendo siempre el contexto de la estrategia y eficiencia en la solución.**

El reto en una entrevista es...

It's your turn to be the Interviewer. The question is available in the left pane below the video



Language: Python ▼

Swap Roles

End Interview

```
1 # logs storage
2 M = [[0 for x in range(n)] for x in range(n)]
3
4
5 # Reset logs
6 def resetLogs():
7     for i in range(0,n):
8         for x in range(0,n):
9             M[i][j] = 0
10
11
12 # Log a call between 2 users
13 def logCall(source, destination):
14     M[source][destination] = M[source][destination] + 1
15
16
17 # get the amount of calls between 2 users
18 def getCallCount(source, destination):
19     print (M[source][destination])
```

The "Call Logger System Design" Problem

You are asked to plan a call log system that logs info about calls between n users.

The system should support the following interface:

- Log a call from user U_x to user U_y .
- Figure how many calls, if at all a given user U_x have made to a given user U_y
- Provide the list of all users that a given user U_x has called to
- Reset all logs and memory of the system

Design the system, choose appropriate data structures and implement the code to support the interface above. Analyze the runtime and space complexity of each interface method you implement.

Solution

The data could be represented by using a 2D matrix of integers of size $n \times n$.

The integer value on a matrix cell $M[x,y]$ will represent the number of calls that user U_x has made to user U_y .

Reset action will be done by writing zeros into all cells. This action will also be done when the system is first initiated:

```
function resetLogs():
    for i from 0 to n-1:
        for j from 0 to n-1:
```

Tips

- Encourage your peer to optimize for of runtime and space complexity combined
- If your peer is stuck, give an example of few calls and ask what would the interface actions do on this data set. Then ask how can you represent this data
- If you peer is still stuck, ask them to choose the most basic representation (even an array for every user on the system) and start improving it

- Proponer una **estrategia eficiente de solución al problema...**
- e **implementarla utilizando las herramientas adecuadas:** algoritmos, estructuras de datos y lenguaje de programación...
- En un **lapso de tiempo**, normalmente no mayor a las **2 horas...**





- Lo que más cuenta para impresionar y conseguir el trabajo es **resolver el problema correctamente**, y tener una **buena estrategia en el proceso de solución...**



**KEEP
CALM
AND BE A
PROBLEM
SOLVER**

Nuestro Reto Semestral...

RESOLVER PROBLEMAS COMO EN UNA ENTREVISTA DE TRABAJO

- En la penúltima semana del semestre, tendremos el **reto final** de resolver un problema de una empresa real...
- Quienes hayan hecho méritos durante el semestre, serán invitados a participar en el reto final...
- Habrá **3 rondas en el semestre** para practicar y generar los méritos para ser invitado al reto final...
- En cada ronda tendremos 2 momentos:
 1. **Momento de PRÁCTICA**, en donde habrá que resolver **cierta cantidad de problemas** antes de una fecha límite. Hay que demostrar que se resolvieron correctamente los problemas para pasar al siguiente momento.
 2. **Momento de SIMULACIÓN**, en donde habrá **un problema a resolver en casa en una fecha y horario específico** (2 horas). Hay que resolver correctamente el problema para pasar generar el mérito de participar en el reto final.
- El **reto final**, tendrá **un problema a resolver** en una fecha y horario específico (2 horas).

Temáticas y fechas de cada RONDA de práctica para el reto

Ronda #1 Problemas de los temas:

Estructuras de datos y algoritmos de búsqueda y ordenamiento.

- Momento de práctica:
 - Fecha límite: *lunes 30 de agosto*
 - Cant. Prob.: 5
 - Tipo Prob.: 2 easy, 2 med y 1 hard
- Momento de simulación: *jueves 2 de septiembre (21:00-23:00)*

Ronda #2 Problemas de los temas:

Algoritmos de Strings, divide y vencerás y PD.

- Momento de práctica:
 - Fecha límite: *lunes 27 de septiembre*
 - Cant. Prob.: 5
 - Tipo Prob.: 2 easy, 2 med y 1 hard
- Momento de simulación: *jueves 30 de septiembre (21:00-23:00)*

Temáticas y fechas de cada RONDA de práctica para el reto

Ronda #3 Problemas de los temas:

Algoritmos de Grafos y Voraces.

- Momento de práctica:
 - Fecha límite: ***lunes 1 de noviembre***
 - Cant. Prob.: **5**
 - Tipo Prob.: 2 easy, 2 med y 1 hard
- Momento de simulación: ***jueves 4 de noviembre (21:00-23:00)***

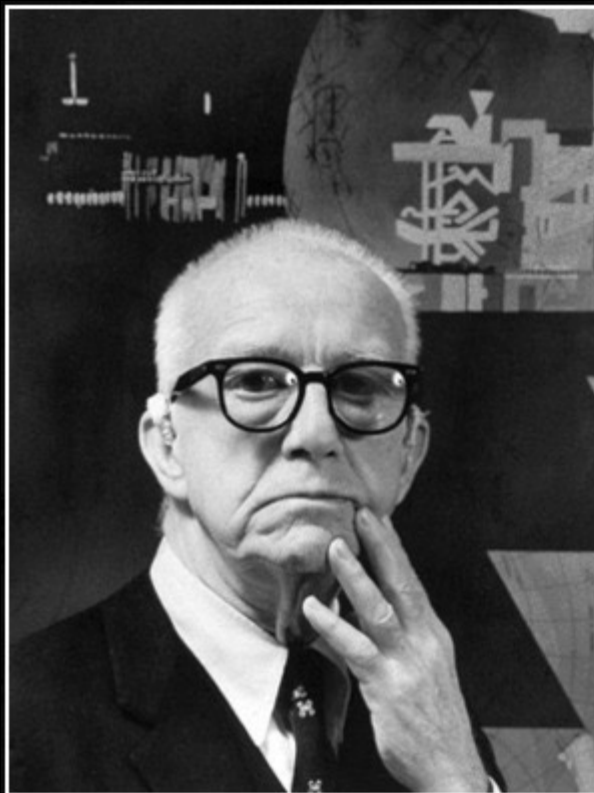
Reto final **Viernes 19 de noviembre a 19:00 hrs**
(esta sesión puede cambiar de fecha y hora,
dependiendo de nuestros invitados)

Valor del reto...

- El valor principal está en el desarrollo de habilidades para cuando tengas una entrevista de trabajo...
- ...así como en la práctica y conexión de lo que ya sabes, con lo nuevo que aprenderás en el curso...
- **5% de la calificación final** se obtiene con los puntos de este reto:
 - En cada una de las 3 rondas:
 - Se obtienen **1 punto** cumpliendo con los problemas del momento de práctica.
 - Si se entregan la cantidad de problemas solicitados resueltos correctamente del momento de práctica, se podrá participar en el momento de simulación, el cuál tiene un valor de **1 puntos**, si se logra resolver el problema asignado.
 - Los méritos para ser invitados a la sesión del reto final son los siguientes:
 - Haber resuelto correctamente TODOS los momentos de practicas de las 3 Rondas y al menos haber resuelto correctamente 2 momentos de simulación
 - En el reto final, se obtendrán **2 puntos** si se logra resolver el problema.
- Con esta política de evaluación, quien participa en todas las oportunidades de las rondas y el reto final, podrá acumular un máximo de 8 puntos, siendo 3 los puntos extras posibles a obtener en la calificación final.
- Después de esta experiencia e independientemente de los puntos que logres, sin duda serás mejor solucionador de problemas con programación... y esto nadie te lo quita...

**Sitio de referencia y en donde se trabajará
para comprobar soluciones**

- <https://vjudge.net/>



The reward for being a good
problem solver is to be heaped with
more and more difficult problems to
solve

— *R. Buckminster Fuller* —

AZ QUOTES

¡Cuidado!
este reto te hará
también más
competitivo para los
concursos de
programación...