

Do Hyun Nam, A01025276

Programación de estructuras de datos y algoritmos fundamentales

11 de octubre del 2021

## Reflexión de la actividad 2.2 (primer acercamiento a la situación problema)

### Programa

```
dohyu@MSI MINGW64 ~/Desktop/SchoolNotes/SEM3/DataStructuresAlgorithms/activities/act2_2 (main)
$ ./a.exe 11
1.- Que direccion ip estas usando?
IP: 192.168.29.11
2.- Cual fue la direccion IP de la ultima conexion que recibio esta computadora? Es interna o externa?
IP: 192.168.29.91      Interna
3.- Cuantas conexiones entrantes tiene esta computadora?
Conexiones entrantes: 1
4.- Cuantas conexiones salientes tiene esta computadora?
Conexiones salientes: 222
5.- Tiene esta computadora 3 conexiones seguidas a un mismo sitio web?
Si hay una conexion seguida a 192.168.29.91
```

### Investigación y reflexión de la importancia y eficiencia del uso de las estructuras de datos lineales

Las estructuras de datos lineales nos permiten iterar sobre sus valores en una sola vez, sin tener que estar empleando otras estructuras auxiliares o iteraciones para reconocer cada valor que los componen. Dada la naturaleza de esta situación problema, es importante considerar que mayoritariamente se están realizando búsquedas sobre la base de datos que se nos proporcionó, y específicamente en esta ocasión utilizar las conexiones entrantes y salientes que un ordenador recibe.

Como ejemplo tomemos las conexiones entrantes que definimos dentro del ADT de conexiones en la computadora. Se emplea un stack para guardar las conexiones entrantes, considerando que dentro de estas conexiones el más primordial a analizar es el más reciente, esta estructura de datos nos ayuda a acceder al dato recién agregado en la primera posición, eliminando la necesidad de iterar sobre la estructura hasta encontrarlo. De esta misma manera, puesto que el ordenador saca conexiones, es importante que se vayan sacando conforme el orden que se van agregando. Si en este caso empleamos un stack, las conexiones que se agregaron al principio nunca saldrían (si es que lo hacen serían muy tardadas), por lo que un queue es la estructura indicada para “formar” y ordenar a los datos conforme se solicita su salida.

Con estos dos ejemplos observamos que las estructuras de datos lineales son de suma utilidad, pero hay que analizar y reconocer el objetivo del reto que se nos presenta para emplear la estructura de datos adecuada para cada única circunstancia.