



# Tecnológico de Monterrey

**Programación de estructuras de datos y algoritmos fundamentales**

**Act 1.3 - Actividad Integral de Conceptos Básicos y Algoritmos Fundamentales  
(Evidencia Competencia)**

**Javier Corona Del Río  
A01023063**

Preguntas:

**¿Cuántos registros tiene tu archivo?**

R= tiene registros 36, 673.

Esta pregunta se leyó el archivo csv con un while y se creo un objeto con las diferentes variables que había en cada línea agregandolos a un vector de objetos. Se imprimió el size con la variable DataBase.size( ) para saber cuántos vectores se crearon y responder la pregunta.

**¿Cuántos records hay del segundo día registrado? ¿Qué día es este?**

R= 3.305. (11-08-2020).

No se ordenó el archivo porque ya venía ordenado pero utilizamos el tipo de búsqueda secuencial search para encontrar la fecha 11-08-2020 y qué se sumen al contador y definir cuantas veces hay registro del segundo día.

**¿Alguna de las computadoras pertenece a Jeffrey, Betty, Katherine, Scott, Benjamin, Samuel o Raymond?**

R= Benjamin se encuentra en la red.

Para esta pregunta se hizo una búsqueda de la variable dentro del vector hostmail\_origin y pasamos las supuestas direcciones de mail de cada persona mencionada en la pregunta ej. nombre.reto.com para identificar en qué posición se encontraba y obviamente si existía. Primero usamos selection sort pero se tardaba aproximadamente 2 a 3 minutos porque se comparaba cada variable con sí mismas entonces optamos por quicksort. En quicksort es más rápido porque se divide el total de variables y las acomoda.

**¿Cuál es la dirección de la red interna de la compañía?**

R= 10.221.242.0

Se usó con un while “-” para qué tomara las direcciones ip y qué con un contador tome las que más se repitan. Estos números son el ip de la compañía porque es el que más se repite (10.221.242) y le agregamos un .0, así, 10.221.242.0 porque en donde está el .0 debe estar la computadora del usuario.

**¿Alguna computadora se llama server.reto.com?**

R= No hay ninguna computadora llamada server.reto.com. (-1)

Se utilizó binary search porque las direcciones donde debería estar server.reto.com estaba ordenado entonces es mucho más eficiente cuando se hiciera la búsqueda al dividirse. Nos dio -1 significando que no existe tal dirección.

### **¿Qué servicio de mail utilizan (algunas ideas: Gmail, Hotmail, Outlook, Protonmail)?**

R= El servicio de mail que utilizan es freemailserver.com

Busco cuantas veces se repetía el ip destino y varias direcciones de mail cómo por ejemplo freemailser.com y lo contamos para ver si ese era el email de la compañía. El que mayor fuera en el contador en este caso freemailserver.com sería la respuesta. Y si, free mail server es el mail linkeado con la ip de la compañía.

### **¿Qué hizo cada quien?**

Para el archivo ReadLines.h:

Kevin y yo definimos la clase Reader lo público y privado. Después yo implementé los getters de las clases que definimos arriba que se utilizaron en las funciones bool. Posteriormente Kevin creó una función para imprimir los objetos del vector DataBase que es el archivo csv.

Para el archivo SortSearch.h:

Ambos implementamos el inicio de la clase padre. Para la tercera pregunta yo había implementado selection sort para responder la pregunta. Servía pero era ineficiente porque tardaba aproximadamente de dos a tres minutos entonces decidimos cambiarlo por algo más eficiente. Kevin hizo la implementación para cambiar de selection a quick sort el cual fue muy efectivo. Más adelante implemente la búsqueda secuencial para obtener la segunda pregunta. Que sirvió a la perfección. Después Kevin implementó la búsqueda secuencial para obtener la tercera y la última pregunta. Para la tercera encontrar si alguno de los nombres mencionados resaltaba y para la última encontrar cuántas veces salía la dirección de mail del ip destino que más se repetía, mostrándonos el resultado. Por último, yo implementé la clase binaria para acomodar en orden alfabético y ver si existe la dirección de server.reto.com, que fue falso.

Para el archivo main.cpp:

Ambos hicimos las funciones bool para la respectiva búsqueda y ordenamiento que realizamos en los headers. Después de bastantes horas terminamos y logramos implementar todo de buena manera respondiendo las cuestiones en el proceso.