

Nombre	Código	RAM	Procesador	GPU
Andres Felipe Vargas Cuadros	202013817	16Gb	Ryzen 7	GTX 1650
Jose Daniel Lozano Moreno	201911107	16Gb	AMD Ryzen 5 3400G	AMD Radeon RX Vega 11 Graphics

- a) Teniendo en cuenta cada uno de los requerimientos ¿Cuántos índices implementaría en el Reto? y ¿Por qué?

Se tendrían 3 índices que despliegan dos subíndices, ya que para el reto se necesitan 3 tipos de algoritmos que se pueden dividir en 2 cada uno. Por lo que esto generaría 6 índices para trabajar al final del reto.

- b) Según los índices propuestos ¿en qué caso usaría Linear Probing o Separate Chaining en estos índices? y ¿Por qué?

1. Para los requerimientos básicos vamos a utilizar SEPARATE CHAINING, esto debido a que se pueden utilizar llaves como los autores que contengan las fechas que van a determinar el orden cronológicamente, esto llevara a tener llaves únicas, pero con valores idénticos.
2. Para los requerimientos intermedios podemos utilizar LINEAR PROBING ya que no se necesita tener un orden cronológico si no solamente clasificar por una llave que puede ser la técnica o la nacionalidad.
3. Para los requerimientos avanzados se pueden usar LINEAR PROBING porque tampoco necesita el orden para determinar una llave en específico.

- c) Dado el número de elementos de los archivos MoMA, ¿Cuál sería el factor de carga para estos índices según su mecanismo de colisión?

El factor de carga seria 0.5 utilizando Linear Probing

- d) ¿Qué diferencias en el tiempo de ejecución notan al ejecutar la cargar los datos al cambiar la configuración de Linear Probing a Separate Chaining?

Para Linear Probing se requiere mas tiempo si la memoria no llega a soportar ya que tendrá que utilizar mas que Separated Chaining ya que solo se busca en el bucket cuando se tiene alguna colisión.

- e) ¿Qué configuración de ADT Map escogería para el índice de técnicas o medios?, especifique el mecanismo de colisión, el factor de carga y el numero inicial de elementos.

```
def keySet(map):
    """
    Retorna una lista con todas las llaves de la tabla de hash

    Args:
        map: El map
    Returns:
        lista de llaves
    Raises:
        Exception
    """
    return ht.keySet(map)
```

Para identificar las llaves únicas que tendrán los valores en colisión.

```
def valueSet(map):
    """
    Retorna una lista con todos los valores de la tabla de hash

    Args:
        map: El map
    Returns:
        lista de valores
    Raises:
        Exception
    """
    return ht.valueSet(map)
```

Para colocar los valores que no tienen colisión en otro lugar.

El mecanismo de colisión sería separate chaining, puesto que es posible que una técnica sea usada por varios artistas, el factor de carga sería de 3.0 y el número inicial sería de 150

- f) ¿Qué configuración de ADT Map escogería para el índice de nacionalidades?, especifique el mecanismo de colisión, el factor de carga y el número inicial de elementos.
- g) El mecanismo de colisión sería el separate chaining, pues es posible que dos obras tengan el mismo índice, el factor de carga sería 4.0 y el número inicial de elementos sería de 194.

```
def contains(map, key):
    """ Retorna True si la llave key se encuentra en el map
    o False en caso contrario.
    Args:
        map: El map a donde se guarda la pareja
        key: la llave asociada a la pareja

    Returns:
        True / False
    Raises:
        Exception
    """
    return ht.contains(map, key)
```

Para identificar las llaves que están en conjunto con otras.

```
def valueSet(map):
    """
    Retorna una lista con todos los valores de la tabla de hash

    Args:
        map: El map
    Returns:
        lista de valores
    Raises:
        Exception
    """
    return ht.valueSet(map)
```

Para colocar los valores que no tienen colisión en otro lugar.