

RETO #2

Juanita Gil Arango –j.gila2@uniandes.edu.co - 202111556

Gabriela Carvajal – g.carvajal@uniandes.edu.co - 202111058

Complejidad temporal:

Req 1:

```
135 def orgartistasCro(catalog, inicial, final):
136     listaArtistas = lt.newList()
137     valores = mp.valueSet(catalog['RangoFechasArtistas'])
138     for artista in valores:
139         if artista['BeginDate']>=inicial and artista['BeginDate']<= final:
140             listaInfo= lt.newList()
141             lt.addLast(listaInfo, artista['DisplayName'])
142             lt.addLast(listaInfo, artista['BeginDate'])
143             lt.addLast(listaInfo, artista['EndDate'])
144             lt.addLast(listaInfo, artista['Nationality'])
145             lt.addLast(listaInfo, artista['Gender'])
146             lt.addLast(listaArtistas, listaInfo)
147     total = lt.size(listaArtistas)
148     return (total, listaArtistas)
```

1.

O(n) (igual en reto 1)

```
150 def listafechas(listaArtistas):
151     listaFechas= lt.newList()
152     for artista in listaArtistas:
153         lt.addLast(listaFechas, artista['BeginDate'])
154     return listaFechas
```

2.

O(n) (no estaba en reto 1)

```
159 def ordenarArtistas(listaOrdenada, listaArtistas):
160     ordenada = lt.newList
161     for fecha in listaOrdenada:
162         for artista in listaArtistas:
163             if fecha == artista['BeginDate']:
164                 lt.addLast(ordenada, artista)
165     return ordenada
```

3.

O(n)² (no estaba en reto 1)

Req 2

```
176 def compararIDayo(catalog, id):
177     #en id entraria el constituent ID del artworks
178     for artist in lt.iterator(catalog['Artist']):
179         if id == artist['ConstituentID']:
180             nomArtista = artist['DisplayName']
181     return nomArtista
```

1.

O(n) (igual en reto 1)

```

191 def orgObrasCro(catalog, inicial, final):
192     listaObras= lt.newList()
193     valores = mp.valueSet(catalog['RangoFechasObras'])
194     for obra in valores:
195         if obra['DateAcquired']>= inicial and obra['DateAcquired']<= final:
196             informacion= lt.newList()
197             lt.addLast(informacion, obra['Title'])
198             lt.addLast(informacion, obra[compararIDayo(catalog['Artist'],obra['ConstituentID'])])
199             lt.addLast(informacion, obra['Date'])
200             lt.addLast(informacion, obra['DateAcquired'])
201             lt.addLast(informacion, obra['Medium'])
202             lt.addLast(informacion, obra['Dimensions'])
203             lt.addLast(listaObras, informacion)
204     conteoObras=lt.size(listaObras)
205     return (conteoObras, listaObras)

```

2.

$O(n)$ (igual en reto 1)

```

207 def listafechasObras(listaObras):
208     listafechas = lt.newList()
209     for obra in listaObras:
210         lt.addLast(listafechas, obra['DateAcquired'])
211     return listafechas

```

3.

$O(n)$ (no estaba en reto 1)

```

213 def ordenarObras(listaOrdenada, listaObras):
214     ordenada = lt.newList
215     for fecha in listaOrdenada:
216         for obra in listaObras:
217             if fecha == obra['DateAcquired']:
218                 lt.addLast(ordenada, obra)
219     return ordenada

```

4.

$O(n)^2$ (no estaba en reto 1)

```

221 def numPurchase(catalog):
222     conteoPu = 0
223     for obra in lt.iterator(catalog['Artworks']):
224         if obra['CreditLine'] == 'Purchase':
225             conteoPu += 1
226     return conteoPu

```

5.

$O(n)$ (igual en reto 1)

Req 3

```

235 def enconID(catalog, nombre: str):
236     encontrarid=None
237     for artista in lt.iterator(catalog['Artists']):
238         if artista['DisplayName']== nombre:
239             encontrarid= artista['ConstituentID']
240             break
241     return encontrarid

```

1.

$O(n)$ (en reto 1: $O(n-1)$)

```

242 def tecnicasartista(catalog, encontrarid):
243     cantidadobras=0
244     tecnicas=lt.newList()
245     for obra in lt.iterator(catalog['Artworks']):
246         if obra['ConstituentID'] == encontrarid:
247             cantidadobras+=1
248             tecnica2= obra['Medium']
249             if tecnica2 in tecnicas:
250                 nuevaObra=lt.newList()
251                 lt.addLast(nuevaObra,obra['Title'])
252                 lt.addLast(nuevaObra,obra['Date'])
253                 lt.addLast(nuevaObra,obra['Medium'])
254                 lt.addLast(nuevaObra,obra['Dimensions'])
255                 lt.addLast(tecnic2,nuevaObra)
256             else:
257                 lt.addLast(tecnicas, tecnica2)
258                 tecnica2=lt.newList()
259                 nuevaObra=lt.newList()
260                 lt.addLast(nuevaObra,obra['Title'])
261                 lt.addLast(nuevaObra,obra['Date'])
262                 lt.addLast(nuevaObra,obra['Medium'])
263                 lt.addLast(nuevaObra,obra['Dimensions'])
264                 lt.addLast(tecnic2,nuevaObra)
265     mayor=0
266     masgrande=None
267     for tec in tecnicas:
268         tamaño= lt.size(tec)
269         if tamaño > mayor:
270             mayor= tamaño
271             masgrande=tec
272     tecnicas= lt.size(tecnicas)
273     tupla=(cantidadobras, tecnicas, masgrande)
274     return tupla

```

2.

2O(n) (en reto 1: O(n))

Req 4

```

288 def idArtists(catalog):
289     for artist in lt.iterator(catalog['Artists']):
290         id = artist['ConstituentID']
291     return id

```

1.

O(n) (igual en reto 1)

```

293 def idyNacio(catalog, id):
294     #en id entraria el constituent ID de artists (return idArtists)
295     for obra in lt.iterator(catalog['Artworks']):
296         if id == obra['ConstituentID']:
297             for artista in lt.iterator(catalog['Artists']):
298                 nacionalidad = artista['Nationality']
299                 mp.put(catalog['Nationality'], nacionalidad, '')

```

2.

O(n)² (igual en reto 1)

```

301 def contNacio(catalog):
302     conteoNa = 0
303     for artist in lt.iterator(catalog['Artists']):
304         nacionalidad = artist['Nationality']
305         natioKeys = mp.keySet(catalog['Nationality'])
306         if nacionalidad in natioKeys:
307             conteoNa += 1
308         mp.put(catalog['Nationality'], nacionalidad, conteoNa)
309     return conteoNa

```

3.

$O(n)$ (igual en reto 1)

```

311 def top10(catalog):
312     valoresNacio=lt.newList()
313     top10= lt.newList()
314     indice = mp.keySet(catalog['Nationality'])
315     pareja = mp.get(catalog['Nationality'], indice)
316     for nacio in pareja:
317         lt.addLast(valoresNacio, nacio)
318     valoresOrdenados= sa.sort(valoresNacio)
319     valorestop10= lt.subList(valoresOrdenados, 1, 10)
320     for valor in lt.iterator(valorestop10):
321         natioKeys = mp.keySet(catalog['Nationality'])
322         for valornacionalidad in natioKeys:
323             if natioKeys[valornacionalidad] == valor:
324                 nacionalidad= valornacionalidad
325                 lt.addLast(top10, nacionalidad)
326     return top10

```

4.

$O(n)^2$ (en reto 1: $O(n+m)$)

```

317 def nacioMasObras(top10, catalog):
318     uno = lt.getElement(top10,1)
319     for artista in lt.iterator(catalog['Artists']):
320         nacionalidad = artista['Nationality']
321         if nacionalidad == uno:
322             id= artista['ConstituentID']
323             for obra in lt.iterator(catalog['Artworks']):
324                 if id == obra['ConstituentID']:
325                     for obra in top10:
326                         x = lt.newList
327                         lt.addLast(x, obra['Title'])
328                         lt.addLast(x, obra['Date'])
329                         lt.addLast(x, obra['Medium'])
330                         lt.addLast(x, obra['Dimensions'])
331                         lt.addLast(x, obra[compararIDayo(obra['ConstituentID'])])
332                     obrasNa = lt.newList
333                     lt.addLast(obrasNa,x)
334     return obrasNa

```

5.

$O(n)^3$ (en reto 1: $O(n)$)

```

336 def lista_nacionalidades(catalog):
337     mayor=0
338     top10= 0
339     lst_nacio_ord = lt.newList
340     natio= mp.keySet(catalog['Nationality'])
341     parejaNatio= mp.get(catalog['Nationality'], natio)
342     for obra in parejaNatio:
343         size= lt.size(obra)
344         if size > mayor:
345             mayor= size
346             while top10 <= 10:
347                 key = mp.valueSet(catalog['Nationality'])
348                 if key == natio['Nationality']:
349                     nacionalidad_mas_repetida = key
350                     top10+= 1
351                 lst_top10_final = lt.addLast(lst_nacio_ord,nacionalidad_mas_repetida)
352     return lst_top10_final

```

6.

O(n) (igual en reto 1)

Req 5

```

355 def obrasDepartamento(departamento, catalog):
356     for obra in lt.iterator(catalog['Artworks']):
357         if catalog['Artworks']['Department'] == departamento:
358             mp.put(catalog['Departamento'],obra['Title'], obra)

```

1.

O(n) (igual en reto 1)

```

359 def listafechas(catalog):
360     listafechas= lt.newList('SINGLE_LINKED')
361     dep= mp.keySet(catalog['Departamento'])
362     parejaDep = mp.get(catalog['Departamento'], dep)
363     for obra in parejaDep:
364         f= obra['Date']
365         t= obra['Title']
366         lt.addLast(listafechas, {'fecha': f, 'titulo': t})
367     return listafechas

```

2.

O(n) (igual en reto 1)

```

373 def listaprecios(catalog):
374     listaprecios= lt.newList('SINGLE_LINKED')
375     obra= mp.keySet(catalog['CostoObras'])
376     parejaObra = mp.get(catalog['Departamento'], obra)
377     for llave in parejaObra:
378         costo= catalog['CostoObras'][llave]
379         lt.addLast(listafechas, {'costo': costo, 'titulo': llave})
380     return listaprecios

```

3.

O(n) (igual en reto 1)

```

386 def pesototal(catalog):
387     peso=0
388     dep= mp.keySet(catalog['Departamento'])
389     parejaDep = mp.get(catalog['Departamento'], dep)
390     for obra in parejaDep:
391         pesoObra= int(obra['Weight'])
392         peso+= pesoObra
393     return peso

```

4.

O(n) (igual en reto 1)

```
399 def dictCostos(catalog):
400     dep= mp.keySet(catalog['Departamento'])
401     parejaDep = mp.get(catalog['Departamento'], dep)
402     for obra in parejaDep:
403         altura=obra['Height']
404         longitud=obra['Length']
405         peso=obra['Weigth']
406         ancho= obra['Width']
407         if (altura== '' or longitud=='') and peso=='':
408             mp.put(catalog['CostoObras'],obra['Title'], 48)
409         else:
410             mayor=0
411             costos=lt.newList()
412             if longitud != '' and altura!= '':
413                 area= (altura*longitud)/100
414                 precioArea= 72/area
415                 lt.addLast(costos, precioArea)
416                 if ancho!='':
417                     volumen= (altura*longitud*ancho)/100
418                     precioVolumen= 72/volumen
419                     lt.addLast(costos, precioVolumen)
420             if peso != '':
421                 precioPeso= 72/peso
422                 lt.addLast(costos, precioPeso)
423             for precio in lt.iterator(costos):
424                 if precio> mayor:
425                     mayor= precio
426             mp.put(catalog['CostoObras'],obra['Title'], mayor)
```

5.

O(n)^2 (igual en reto 1)

```
434 def obrasMasAntiguas(listaOrdenada, catalog):
435     x= lt.subList(listaOrdenada, (lt.size(listaOrdenada))-4, 5)
436     masAntiguas= lt.newList()
437     for obra in lt.iterator(x):
438         info= lt.newList()
439         lt.addLast(info, obra['Title'])
440         id= obra['ConstituentID']
441         artista = compararIDayo(catalog,id)
442         lt.addLast(info, artista)
443         lt.addLast(info, obra['Classification'])
444         lt.addLast(info, obra['Date'])
445         lt.addLast(info, obra['Medium'])
446         lt.addLast(info, obra['Dimensions'])
447         costotransporte= dictCostos(catalog)
448         for llave in catalog:
449             if llave == obra['Title']:
450                 costo=costotransporte[llave]
451                 break
452         lt.addLast(info, costo)
453         lt.addLast(masAntiguas, info)
454     return masAntiguas
```

6.

$2O(n)^2$ (igual en reto 1)

```
456 | def obrasMasCost(listaOrdenadaprecios2, catalog):
457 |     x= lt.subList(listaOrdenadaprecios2, (lt.size(listaOrdenadaprecios2))-4, 5)
458 |     masCost= lt.newList()
459 |     for obra in lt.iterator(x):
460 |         info= lt.newList()
461 |         lt.addLast(info, obra['Title'])
462 |         id= obra['ConstituentID']
463 |         artista = compararIDayo(catalog, id)
464 |         lt.addLast(info, artista)
465 |         lt.addLast(info, obra['Classification'])
466 |         lt.addLast(info, obra['Date'])
467 |         lt.addLast(info, obra['Medium'])
468 |         lt.addLast(info, obra['Dimensions'])
469 |         for llave in catalog['Departamento']:
470 |             if llave == obra['Title']:
471 |                 costo=catalog['Departamento'][llave]
472 |                 break
473 |         lt.addLast(info, costo)
474 |         lt.addLast(masCost, info)
475 |     return masCost
```

7.

$O(n)^2 + O(n)$ (igual en reto 1)

Requerimientos individuales:

Req. 3: Gabriela Carvajal

Req. 4: Juanita Gil