

10)

## Bucket Sort ( $A, N$ )

```
1 cria  $N$  listas  $B[i]$   
2 para  $i \leftarrow -1$  até  $N$  faça  
3    $B[i] \leftarrow \text{NIL}$   
4 para  $i \leftarrow -1$  até  $N$  faça  
5   insira ( $B[\text{floor}(A[i]/N)]$ ,  $A[i]$ )  
6 para  $i \leftarrow -1$  até  $N$  faça  
7   ordena ( $B[i]$ )  
8  $A \leftarrow \text{concatena Listas}(B, N)$   
9 return  $A$ 
```

O bucket sort é capaz de ordenar a coleção dada em tempo linear porque usamos a divisão inteira de  $x \in [0, n^2 - 1]$  por  $N$ . A linearidade é, de fato, observada quando se trata da ordenação das  $N$  listas, porque se a coleção for espalhada podemos fazê-la em  $O(1)$ .

11) Usando  $A = ac - bd$  e  $B = ad + bc$ , podemos considerar as seguintes multiplicações  $M_1 = ac$ ,  $M_2 = bd$  e  $M_3 = (a+b)(c+d)$ , tal que  $A = M_1 - M_2$  e  $B = M_3 - M_1 - M_2$ . Assim, verifica-se que foram usadas apenas 3 multiplicações reais.

MultComplex(a, b, c, d)

1.  $M_1 = a \cdot c$

2.  $M_2 = b \cdot d$

3.  $M_3 = (a+b)(c+d)$

4. return  $[M_1 - M_2, M_3 - M_1 - M_2]$