

2) Toma de tiempos de ejecución

Ejercicio 1: Esta forma de contar la podremos usar durante $5,8 \cdot 10^8 - 56$ años.

Ejercicio 2:

Ejercicio 3: el tiempo medido a veces sale 0 debido a factores como la velocidad del procesador que ejecuta el código en un tiempo menor a 1 milisegundo.

También puede ser porque si los valores t_2 y t_1 pueden ser similares, al hacer la resta el resultado queda aproximadamente en 0.

Ejercicio 4: el tamaño n del problema donde los tiempos empiezan a ser fiables (mayores a 50ms) son aproximadamente 8 millones.

2) Crecimiento del tamaño del problema

Tiempos de ejecución de Vector 3

```
C:\Users\jaime\Desktop\alg_GonzalezDiazJaime306054\p11>java -Xint p11.Vector3
n    tTiempo
10000    0
20000    0
40000    1
80000    1
160000   1
320000   3
640000   7
1280000  13
2560000  23
5120000  56
10240000 104
20480000 187
40960000 375
81920000 804
```

4) Toma de tiempos pequeños (<50ms)

Ejercicio 1: tiempos del vector 4 con 1, 10, 1000

```
C:\Users\jaime\Desktop\alg_GonzalezDiazJaime306054\p11>java -Xint Vector4.
java 1
repeticiones = 1
n      Tiempo
10000  0
20000  0
40000  0
80000  1
160000 1
320000 2
640000 6
1280000 8
2560000 18
5120000 36
10240000 71
20480000 137
40960000 286
81920000 578
Fin de la medicion de tiempos *****
```

```
C:\Users\jaime\Desktop\alg_GonzalezDiazJaime306054\p11>java -Xint Vector4.
java 10
repeticiones = 10
n      Tiempo
10000  0
20000  2
40000  4
80000  6
160000 13
320000 23
640000 42
1280000 85
2560000 174
5120000 356
10240000 705
20480000 1401
40960000 2826
81920000 5655
Fin de la medicion de tiempos *****
```

```
C:\Users\jaime\Desktop\alg_GonzalezDiazJaime306054\p11>java -Xint Vector4.
java 100
repeticiones = 100
n      Tiempo
10000  7
20000  13
40000  29
80000  55
160000 112
320000 222
640000 438
1280000 880
2560000 1753
5120000 3534
10240000 7025
20480000 14098
40960000 28073
81920000 56841
Fin de la medicion de tiempos *****
```

Ejercicio 1: Si el tamaño del problema se duplica por dos, el tiempo de ejecución también se duplica.

Ejercicio 2: Si el tamaño del problema se multiplica por k, entonces el tiempo de ejecución también aumenta proporcionalmente a k.

Ejercicio 3: Sí, los tiempos obtenidos (reflejados en Tsuma) son esperados con la complejidad lineal $O(n)$.

Estos tiempos se han medido en un ordenador con procesador Intel i7-12700H y con una memoria RAM de 16GB.

n	Tsuma	Tmaximo
10000	67	86
20000	132	176
40000	266	362
80000	551	736
160000	1072	1468
320000	2212	2826
640000	4403	5644
1280000	8731	11275
2560000	17578	22538
5120000	35255	57359
10240000	FdT	FdT
20480000	FdT	FdT
40960000	FdT	FdT
81920000	FdT	FdT

n	Tcoincidencias1	Tcoincidencias2
10000	877	124
20000	3475	202
40000	13834	416
80000	55481	807
160000	FdT	2180
320000	FdT	6264
640000	FdT	9735
1280000	FdT	17637
2560000	FdT	38641
5120000	FdT	FdT
10240000	FdT	FdT
20480000	FdT	FdT
40960000	FdT	FdT
81920000	FdT	FdT

Los tiempos sí encajan con lo esperado de la complejidad de cada código.