

INFO2050: Sort

Julien GUSTIN, Mathias CARLISI

Table des matières

1	Analyse théorique	3
1.1	invariant	3
1.2	PseudoCode	4
1.3	Complexité	4
1.4	Stabilité ?	5
1.5	Complexité au pire cas	6
2	Analyse expérimentale	7
2.1	Temps d'exécution sur des tableaux aléatoires	7

1 Analyse théorique

1.1 invariant

1.2 PseudoCode

NEW-SORT(A)

```
1   $i = 1$ 
2  for  $j = 2$  to  $A.length$ 
3      while  $A[j] < A[j + 1]$  and  $j < A.length - 1$ 
4           $j = j + 1$ 
5      MERGE( $A, 1, i, j$ )
6   $i = j$ 
```

1.3 Complexité

La complexité en temps dans le meilleur des cas est de $\Theta(n)$ et dans le pire des cas $\Theta(n^2)$

1.4 Stabilité ?

1.5 Complexité au pire cas

n	InsertionSort	QuickSort	HeapSort	MergeSort	NewSort
10^1	0,000015	0,000005		0,000009	0,000005
10^2	0,000052	0,000035		0,000054	0,000090
10^3	0,001055	0,000225		0,000297	0,001841
10^4	0,060314	0,003429		0,002229	0,112815
10^5	7,216459	0.036962		0,017463	14,224125
10^6	768,522644	3.239571		0,179450	1529,154037

2 Analyse expérimentale

2.1 Temps d'exécution sur des tableaux aléatoires