

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   $i = 0$ 
2  for  $j = 1$  to  $A.length$ 
3      while  $A[j] < A[j + 1]$  and  $j < A.length - 1$ 
4           $j = j + 1$ 
5      if  $i > 0$ 
6          MERGE( $A, 1, i, j$ )
7   $i = j$ 
```

1.3 Complexité

1.4 Stabilité ?

1.5 Complexité au pire cas

| n | InsertionSort | QuickSort | HeapSort | MergeSort | NewSort |
|----|---------------|-----------|----------|-----------|---------|
| 10 | 0,000015 | | | | |
| 10 | 0,000052 | | | | |
| 10 | 0,001055 | | | | |
| 10 | 0,060314 | | | | |
| 10 | 7,216459 | | | | |
| 10 | | | | | |

2 Analyse expérimentale

2.1 Temps d'exécution sur des tableaux aléatoires