

```
mergeSort(ITEM[] A, int primo, int ultimo)
```

```
if primo < ultimo then // devono esistere almeno due elementi
    int mezzo ← ⌊  $\frac{\text{primo} + \text{ultimo}}{2}$  ⌋
    mergeSort(A, primo, mezzo)
    mergeSort(A, mezzo+1, ultimo)
    merge(A, primo, ultimo, mezzo)
```

```
merge(ITEM A, int primo, int ultimo, int mezzo)
```

```
int i, j, k, h
// inizializzo i puntatori
i ← primo j ← mezzo k ← primo
// k: indica la prossima posizione di scrittura
// fintanto che entravi
while i ≤ mezzo and j ≤ ultimo do
    if A[i] ≤ A[j] then
        // l'elemento è già ordinato
        B[k] ← A[i]
        i ++
    else
        B[k] ← A[j]
        j ++
    // in entrambi i casi ho inserito un valore
    k ++
// se uno dei due vettori finisce ricopio la parte ordinata alla fine del vettore
d'appoggio
j ← ultimo
da h ← mezzo fino a i fai
    A[j] ← A[h]
    j --
// ricopio il vettore d'appoggio del vettore originale
da j ← primo fino a k - 1 fai
    A[j] ← B[j]
```