
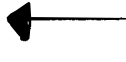


## Exercise 2

Prove Mergesort.

```
01: vector<int> mergesort_recv(vector<int> arr, int lo, int hi){
02:   vector<int> answer;
03:   int mid;
04:   if(hi-lo <= 1) {
05:     answer.push_back(arr[lo]);
06:     return answer;
07:   }
08:   mid = floor((lo+hi)/2);
09:   vector<int> left = mergesort_recv(arr, lo, mid);
10:   vector<int> right = mergesort_recv(arr, mid, hi);
11:   answer = merge_desc(left, right);
12:   return answer;
13: }
```

- 1) Δεδομένο ότι ο merge λειτουργεί ορθά  
- 2) Θεωρώ πίνακα 'α' τον αρχικό πίνακα

Βήμα 1<sup>ο</sup> : Βάση.

για  $n=1$ ,  $hi-lo = 0 \leq 1$

- Συνεπώς θα επιστρέψει πίνακας μεγέθους 1 με το μοναδικό element της α, από γραμμή 4-6.
- Κάθε πίνακας  $n \leq 1$  είναι ταξινομημένος, καθώς δεν υπάρχει ζεύγος  $i, j$  με  $i=j+1$  για το οποίο να ισχύει  $\alpha[i] > \alpha[j]$

Βήμα 2<sup>ο</sup> : Επαγωγική υπόθεση (ισχύει επαγωγικά).

- Mergesort λειτουργεί ορθά για κάθε  $n < n \Rightarrow$   
Mergesort λειτουργεί ορθά για  $n$

1 > Έστω πίνακας α μεγέθους  $n$

2 > Έστω το πρώτο αναδρομικό στάδι του Mergesort στον πίνακα α μεγέθους  $n$ .

3 > Στο πρώτο στάδι, θα ισχύει  $n=n$  και  $L=0$

4 > Από γραμμή 8  $mid = (lo+hi)/2$

5 > Συνεπώς, στην γραμμή 9 του πρώτου αναδρομικού σταδίου, είτε έχουμε Mergesort σε πίνακα μεγέθους  $\alpha_1 = mid-lo$

6 > Επόμενον, στην γραμμή 10 του πρώτου αναδρομικού σταδίου, είτε έχουμε Mergesort σε πίνακα μεγέθους  $\alpha_2 = hi-mid$

7 > Από 4, ισχύει  $\alpha_1 < n$  και  $\alpha_2 < n$ .

8 > Αρα από ισχυρή υπόθεση, ο Mergesort στον  $\alpha_1$  και  $\alpha_2$  έχει επιστρέψει ορθά αποτελέσματα

9 > Οι left και right από 8 είναι ταξινομημένοι

10 > από 9 και δεδομένο 1 (merge λειτουργεί ορθά) η μεταβλητή answer στην γραμμή 11 περιλαμβάνει έναν ταξινομημένο πίνακα μεγέθους  $n$

11 > Γραμμή 12 επιστρέφει τον ταξινομημένο answer μεγέθους  $n$

12 > Από 11 ο Mergesort επιστρέφει τον α μεγέθους  $n$ , ταξινομημένο

13 > Από 12 ο Mergesort είναι ορθός.