

Algoritmi sortiranja

Merge Sort i Quick Sort

Vježbe 4

Merge Sort

Primjer python implementacije:

```
def merge_sort(niz):  
    if len(niz) <= 1:  
        return niz  
    pola = len(niz)//2  
    desno = niz[pola:]  
    levo = niz[:pola]  
    levo_sortirano = merge_sort(levo)  
    desno_sortirano = merge_sort(desno)  
    sortiran = spoj(levo_sortirano,desno_sortirano)  
    •    return sortiran
```

Merge Sort

```
def spoj(levo_sortirano, desno_sortirano):  
    levi_indeks, desni_indeks = 0, 0  
    kombinacija = []  
    while levi_indeks < len(levo_sortirano) and desni_indeks <  
len(desno_sortirano):  
        if levo_sortirano[levi_indeks] < desno_sortirano[desni_indeks]:  
            kombinacija.append(levo_sortirano[levi_indeks])  
            levi_indeks += 1  
        else:  
            kombinacija.append(desno_sortirano[desni_indeks])  
            desni_indeks += 1  
    kombinacija.extend(desno_sortirano[desni_indeks:])  
    kombinacija.extend(levo_sortirano[levi_indeks:])  
    •    return kombinacija
```

Quick Sort

Primjer python implementacije:

```
def quick_sort(niz):  
    if len(niz) <= 1:  
        return niz  
    pivot = niz[-1]  
    manji, jednaki, veci = podijeliPoVelicini(niz,  
        pivot)  
    •    return quick_sort(manji) + jednaki +  
        quick_sort(veci)
```

Quick Sort

```
def podijeliPoVelicini(niz, pivot):  
    manji, jednaki, veci = [],[],[]  
    for el in niz:  
        if el < pivot:  
            manji.append(el)  
        elif el == pivot:  
            jednaki.append(el)  
        else:  
            veci.append(el)  
    •    return manji, jednaki, veci
```