

Ασκηση 1 εργασία 2:

Για τη sort:

Εστω T ο αλγοριθμος ολοκληρης της συναρτησης sort. Εστω επισης T1 ο αλγοριθμος της πρωτης επαναληψης for-loop (στη γραμμη 17) , εστω T2 ο αλγοριθμος της δευτερης επαναληψης for-loop (στη γραμμη 25).

Ισχυει οτι $T = T1 + T2$.

Η πολυπλοκοτητα του αλγοριθμου T εξαρταται αποκλειστικα απο την μεγαλυτερη αναμεσα στα T1 και T2.

$$\text{Για } T1: T1(n) = \sum_{i=1}^n 1 = 1+1+1+\dots+1 = N$$

$$\text{Για } T2: T2(n) = \sum_{i=1}^n \left(\sum_{j=i}^n 1 \right) = (1+2+3+\dots+n) = n*(n+1)/2 = N*N$$

Αρα η πολυπλοκότητα του T2 είναι $N*N$ ενώ του T1 είναι N. Οπότε $T = N*N$

Για τη main:

Εστω T ο αλγοριθμος ολοκληρης της main. Εστω επισης T1 ο αλγοριθμος της πρωτης επαναληψης for-loop (στη γραμμη 43) , εστω T2 ο αλγοριθμος της δευτερης επαναληψης for-loop (στη γραμμη 46) , εστω T3 ο αλγοριθμος της τριτης επαναληψης for-loop (στη γραμμη 52) , και εστω Tsort ο αλγοριθμος της συναρτησης sort που βρηκαμε προηγουμενως.

Ισχυει οτι $T = T1 + T2 + T3 + T4 + Tsort$.

Η πολυπλοκοτητα του αλγοριθμου T εξαρταται αποκλειστικα απο την μεγαλυτερη αναμεσα στα T1 , T2 , T3 , T4 , Tsort.

Για ολα τα T1 , T2 , T3 , T4 ισχυει οτι:

$$T1(n) = \sum_{i=0}^n 1 = 1+1+1+\dots+1 = N$$

Για $Tsort(n) = N*N$

Αρα η πολυπλοκότητα του Tsort είναι $N*N$ ενώ των T1 , T2 , T3 , T4 είναι N. Οπότε $T_{main} = N*N$