Sortera resultatlistan

Du får givet poängen för flera deltagare i en tävling. Din uppgift är att skapa en resultatlista för deltagarna, sorterad i sjunkande ordning efter poängen.

Olyckligtvis stödjer datastrukturen som används för listan endast en operation, vilken flyttar en deltagare från position i till position j utan att ändra de övriga spelarnas inbördes ordning. Om $i > j$, ökar alltså positionerna för deltagarna på positionerna mellan j och $i - 1$ med 1, medan om $i < j$ så minskar positionerna för deltagarna på positionerna mellan $i + 1$ och j med 1.

Denna operation tar i steg för att hitta deltagaren som ska flyttas och j steg för att hitta positionen dit han eller hon ska flyttas, så den totala kostnaden för att flytta en deltagare från position i till position j är $i + j$. Positionerna är numrerade så att de startar med 1.

Bestäm en sekvens av flyttningar som leder till den sorterade resultatlistan på ett sådant sätt att summan av kostnaderna för flyttningarna minimeras.

Indata

Indatan läses från en textfil med namnet `sorting.in`. Den första raden innehåller n ($2 \leq n \leq 1000$), antalet spelare. Var och en av de följande n raderna innehåller ett icke-negativt heltal s_i ($0 \leq s_i \leq 1{,}000{,}000$), poängen för deltagarna i den ursprungliga ordningen. Du kan anta att alla poängtal är olika.

Utdata

Utdatan skrivs till en textfil med namnet `sorting.out`. På första raden av utdatafilen, skriv antalet flyttningar som använts för att skapa resultatlistan. De följande raderna ska specificera flyttningarna i den ordning de utförs. Varje flyttning ska beskrivas genom en rad med två heltal i och j, vilka betyder att deltagaren på position i flyttas till position j. Talen i och j måste separeras av ett enda blanksteg.

Exempel

sorting.in	sorting.out
5 20 30 5 15 10	2 2 1 3 5

Poängsättning

30% av testfallen har $n \leq 10$