Sort by swaps

Problem Code: **SWPSRT1**

Submit (/PEC2021E/submit/SWPSRT1)

Swap Sort

You are given an array A of size n (not exceeding 500), you are allowed a maximum of $n-1$ swaps where in each swap you can choose **any two** indices i, j such that $0 \leq i, j \leq n-1$ and swap the values of A_i and A_j.

Note that you **do not** need to minimise the number of swaps, and you are also **not required** to optimize the time complexity (note that the constraints allow solutions that are $\mathcal{O}(n^2)$ as well), however you **cannot** swap more than $n-1$ times.

There may be many solutions, you can output **any** of them.

Input

The first line contains a single integer n, the number of elements in the array

The second line contains n space separated integers, the elements of the array

Output

In the first line print the number of swaps you need to perform, let this number be m.

In the next m lines, print two space separated integers i, j such that $0 \leq i, j \leq n-1$.

After performing the swaps in the order in which your program gives the output, the array should become sorted in increasing order. (i.e. $A_i \geq A_{i-1}$ for every i in the range $[1, n-1]$).

Constraints

$1 \leq n \leq 500$

$-10^9 \leq A_i \leq 10^9$

Sample Input

```
3
3 2 1
```

Sample Output

```
1
0 2
```

Explanation

After the first swap, the elements A_0 and A_2 are swapped, and the array becomes sorted, i.e. $\{1, 2, 3\}$

Author:	srikkanth_adm (/users/srikkanth_adm)
Date Added:	20-02-2021
Time Limit:	1 secs
Source Limit:	50000 Bytes
Languages:	CPP14, C, JAVA, PYTH 3.6, PYTH, CS2, ADA, PYPY, PYP3, TEXT, PAS fpc, RUBY, PHP, NODEJS, GO, TCL, HASK, PERL, SCALA, kotlin, BASH, JS, PAS gpc, BF, LISP sbcl, CLOJ, LUA, D, R, CAML, rust, ASM, FORT, FS, LISP clisp, SQL, swift, SCM guile, PERL6, CLPS, WSPC, ERL, ICK, NICE, PRLG, ICON, PIKE, COB, SCM chicken, SCM qobi, ST, NEM, SQLQ

Submit (/PEC2021E/submit/SWPSRT1)

Comments