

# (Challenge) Chef and Palindromes

Problem Code: CHEFPALS



Chef loves palindromes! Chef loves them so much he can see palindromes anyw here! Even in this problem.

Chef has  $N$  sequences of white cells. For each  $i$  ( $1 \leq i \leq N$ ), the  $i$ -th sequence is formed by  $l_i$  cells; Chef has to paint exactly  $b_i$  of these cells black.

After painting those cells, Chef writes down the lengths of all white blocks (maximum contiguous subsequences of white cells) from left to right as a single string of digits.

For example, if Chef's initial sequence of cells is "wwwwwwww" and he has to paint three cells black, he can get "bw bw bw ww" ('w' stands for a white cell, 'b' for a black cell). In this case, he will write down the lengths of white blocks 2, 1, 3 as the string "213". He can also paint it to get "bbwwwwbw" and write down "51". He can even paint it to get "bbbw www" and write down "6". There are many other options as well.

After painting all sequences, Chef will have written down a list of  $N$  strings. Now he can reorder these strings **in any way he wants** and concatenate them into one string  $S$ .

It's time for palindromes! Chef wants the resulting string  $S$  to be as close to a palindrome as possible. The distance of  $S$  to a palindrome is defined as the minimum number of characters in  $S$  that Chef needs to change to get a palindrome.

For example, the string "123345" has distance 2 to a palindrome. Strings "2" and "1221" have distance 0 to a palindrome, since they are already palindromes.

You need to find a way to paint some cells, reorder and concatenate the resulting strings to form a string  $S$  with the minimum possible distance to a palindrome.

---

## Input

- The first line of the input contains a single integer  $N$ .
- The second line contains  $N$  space-separated integers  $l_1, l_2, \dots, l_N$ .
- The third line contains  $N$  space-separated integers  $b_1, b_2, \dots, b_N$ .

---

## Output

- The first  $N$  lines of the output should describe the sequences of cells obtained after painting some cells black.
- The  $i$ -th of these lines should contain a single string of length  $l_i$  consisting of letters 'b' and 'w'. There should be exactly  $b_i$  letters 'b' in this string.
- The following line should contain a sequence of  $N$  space-separated integers  $p_1, p_2, \dots, p_N$  denoting the order of strings written down for each sequence of cells — the  $i$ -th string in the concatenation should be the string written down for the  $p_i$ -th sequence of cells.  $p_1, p_2, \dots, p_N$  has to be a permutation of numbers 1 through  $N$ .

---

## Constraints

- $N = 1000$
- $1 \leq l_i \leq 1000$  for each valid  $i$
- $0 \leq b_i < l_i$  for each valid  $i$

---

## Example

**Input :**

```
4
5 10 3 8
2 2 1 2
```

**Output :**

```
wwbbw
wwwbbbwww
wwb
wwwbbbw
2 1 3 4
```

---

## Explanation

- In the first sequence of cells, painting the third and fourth cell gives "**wwbbw**" with white blocks of lengths 2 and 1, which gives Chef the string "**21**".
- "**wwwbbbwww**" gives Chef the string "**44**".
- "**wwb**" gives Chef "**2**".
- "**wwwbbbw**" gives Chef "**42**".

After reordering the strings "21", "44", "2", "42" using the permutation (2, 1, 3, 4), Chef gets strings "44", "21", "2", "42" and concatenates them to **S** = "**4421242**". The distance of this string to a palindrome is 1, as Chef only has to replace the last digit with '4' to get a palindrome.

---

## Scoring

- If your output is invalid in at least one test case, you will receive WA verdict.
- Your score for each test case is equal to the distance of the string **S** (obtained for your output) to a palindrome plus 1.
- The total score for a submission is the sum of scores for all test cases.
- Your goal is to minimise the total score.
- During the contest, your score will be calculated only for one test case from each of groups 1, 3, 5 and 7 of test cases. After the contest, there will be a rejudge using all test cases from all groups.

---

## Test generation

- Let  $\text{random}(x)$  denote a uniformly random integer in the range  $[0, x]$ .
- For each valid  $i$ ,  $l_i = \text{random}(999) + 1$ .
- There will be 8 groups of test cases, groups 1, 3, 5 have 3 test-cases each, and

groups 2, 4 have 4 test-cases each, while groups 6, 7, 8 have one test-case each.

- o In group 1,  $b_i = \text{random}(\text{random}(l_i - 1))$  for each valid  $i$ .
- o In group 2,  $b_i = \text{random}(\text{random}(\text{random}(l_i - 1)))$  for each valid  $i$ .
- o In group 3,  $b_i = \text{random}(\text{random}(\text{random}(\text{random}(l_i - 1))))$  for each valid  $i$ .
- o In group 4,  $b_i = \text{random}(\text{random}(\text{random}(\text{random}(\text{random}(l_i - 1)))))$  for each valid  $i$ .
- o In group 5,  $b_i = \text{random}(\text{random}(\text{random}(\text{random}(\text{random}(\text{random}(l_i - 1)))))$  for each valid  $i$ .
- o In group 6,  $b_i = \text{random}(\text{random}(\text{random}(\text{random}(\text{random}(\text{random}(\text{random}(l_i - 1)))))$  for each valid  $i$ .
- o In group 7,  $b_i = \text{random}(\text{random}(\text{random}(\text{random}(\text{random}(\text{random}(\text{random}(\text{random}(l_i - 1)))))$  for each valid  $i$ .
- o In group 8,  $b_i = \text{random}(\text{random}(\text{random}(\text{random}(\text{random}(\text{random}(\text{random}(\text{random}(l_i - 1)))))$  for each valid  $i$ .

Author: [berezin \(/users/berezin/\)](/users/berezin/)

Date Added: 12-10-2015

Time Limit: 1 secs

Source Limit: 50000 Bytes

Languages: ADA, ASM, BASH, BF, C, CAML, CLOJ, CLPS, CPP 4.3.2, CPP 6.3, CPP14, CS2, D, ERL, FORT, FS, GO, HASK, ICK, ICON, JAVA, JS, kotlin, LISP clisp, LISP sbcl, LUA, NEM, NICE, NODEJS, PASC fpc, PASC gpc, PERL, PERL6, PHP, PIKE, PRLG, PYPY, PYTH, PYTH 3.5, RUBY, rust, SCALA, SCM chicken, SCM guile, SCM qobi, ST, swift, TCL, TEXT, WSPC

---