Problem B. Buffcraft

Input file:	buffcraft.in
Output file:	buffcraft.out
Time limit:	2 seconds
Memory limit:	256 megabytes

Brenda enjoys a new role-playing game *Buffcraft*. Shields, swords, books and other carry-on items do not affects character stats in *Buffcraft*. The only way to increase the stats of your character is to buff her.

There are two types of buffs in *Buffcraft*. Direct buffs increase a base value of the stat, while percentage buffs increase stats by the fraction of the base value. To be precise, if unbuffed base value of your character stat is b, you have buffed her using n direct buffs of strength $d_1, d_2, ...d_n$ and m percentage buffs of strength $p_1, p_2, ..., p_m$, the resulting stat will be equal to $(b + d_1 + d_2 + \cdots + d_n)(100 + p_1 + p_2 + \cdots + p_m)/100$. Note that the resulting stat may be fractional.

Unfortunately, your character has only k buff slots and if you apply more than k buffs on her, only the last k buffs remains active. Thus, there is no reason to apply more than k buffs simultaneously. You cannot apply the same buff more than once.

Brenda is going to send his character to raid and wants to buff her health to maximal possible value. She has some direct and some percentage buffs at her disposal and needs your help to select the set of buffs that leads to maximal possible total health.

Input

The first line of the input file contains four integers b, k, c_d and c_p — the base health of the character, the number of buff slots, the number of available directs buffs, and the number of available percentage buffs.

The following line contains c_d integers d_i — strengths of direct buffs.

The last line of the input file contains c_p integer numbers p_i — strengths of percentage buffs.

All numbers in the input file are greater than or equal to zero, and less than or equal to fifty thousand.

Output

The first line of the output file must contain two integers n and m — the number of direct and percentage buffs to use ($0 \le n \le c_d$; $0 \le m \le c_p$; $0 \le n + m \le k$).

The following line must contain n different numbers — indices of direct buffs to apply (buffs are numbered from one).

The last line of the output must contain m different numbers — indices of percentage buffs to apply (also numbered from one).

The resulting total health after application of all $n + m$ buffs must be maximal possible.

Examples

buffcraft.in	buffcraft.out
70 3 2 2 40 30 50 40	2 1 2 1 1
1 2 3 4 6 6 5 8 10 7 9	2 0 1 2