# Problem 4 – Five Special Letters

We are given two numbers: **start** and **end**. Write a program to **generate all sequences of 5 letters**, each from the set { '**a**', '**b**', '**c**', '**d**', '**e**' }, such that the weight of these 5 letters is a number in the range [**start** … **end**] inclusively. Print them in alphabetical order, in a single line, separated by a space.

The **weight of a single letter** is calculated as follows: weight('**a**') = **5**; weight('**b**') = **-12**; weight('**c**') = **47**; weight('**d**') = **7**; weight('**e**') = **-32**. The **weight of a sequence** of letters c1c2…cn is the calculated by first removing all repeating letters (from right to left) and then calculate the formula:

weight(c1c2…cn) = 1\*weight(c1) + 2\*weight(c2) + … + n\*weight(cn)

For example, the weight of "**bcddc**" is calculated as follows: First we remove the repeating letters and we get "**bcd**". Then we apply the formula: 1\*weight('**b**') + 2\*weight('**c**') + 3\*weight('**d**') = 1\*(-12) + 2\*47 + 3\*7 = 103. Another example: weight("cadea") = weight("cade") = 1\*47 + 2\*5 + 3\*7 - 4\*32 = -50.

### Input

The input data should be read from the console. It will consist of 2 lines:

* The number **start** stays at the first line.
* The number **end** stays at the second line.

The input data will always be valid and in the format described. There is no need to check it explicitly.

### Output

The output should be printed on the console as a sequence of strings in **alphabetical order**. Each string should be separated than the next string by a single space. In case no 5-letter strings exist with a weight in the specified range, print “**No**”.

### Constraints

* The numbers **start** and **end** will be an **integers** in the range [-10000…10000].
* Allowed working time for your program: 0.25 seconds.
* Allowed memory: 16 MB.

### Examples

|  |  |  |
| --- | --- | --- |
| **Input** | **Output** | **Comments** |
| 40  42 | bcead bdcea | weight("bcead") = 41  weight("bdcea") = 40 |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Input** | **Output** |  | **Input** | **Output** |  | **Input** | **Output** |
| -1  1 | bcdea cebda eaaad eaada eaadd eaade eaaed eadaa eadad eadae eadda eaddd eadde eadea eaded eadee eaead eaeda eaedd eaede eaeed eeaad eeada eeadd eeade eeaed eeead | 200  300 | baadc babdc badac badbc badca badcb badcc badcd baddc bbadc bbdac bdaac bdabc bdaca bdacb bdacc bdacd bdadc bdbac bddac beadc bedac eabdc ebadc ebdac edbac | 300  400 | No |