

Problem 2: NOMAD is an Island

NOMAD is an autonomous deep-space probe whose actions are guided by one of a provided set of command directives. In the event of an accident that corrupts main memory by erasing one or more bits of the selected directive, NOMAD must compare the corrupted directive against the backup directive set and decide which directives in the backup set might possibly have been the selected directive. The comparison criterion is that a corrupted bit (denoted by ‘?’) can match any bit and that the total number of mismatches of uncorrupted ‘0’/‘1’ bits must be $\leq k$ for some specified error threshold k . For example, given a corrupted directive ‘10?01’ and ordered backup directive set \langle ‘10101’, ‘10001’, ‘10000’, ‘10010’, ‘11010’, ‘01010’ \rangle , backup directives 1, 2, 3, and 4 are compatible with the corrupted directive when $k = 2$ and backup directives 1 and 2 are compatible when $k = 0$.

Write a program which, given a corrupted directive, a list of n backup directives, and an error threshold k , computes and outputs all backup directives that are k -compatible with the corrupted directive. If there are no k -compatible backup directives, print an appropriate message. All directives will be the same length and not contain any whitespace characters and a corrupted directive will contain at least one corrupted bit. Your input will be an $(n + 2)$ -line textfile, in which the first line gives the corrupted directive, the second line gives the values of n and k , and each of the remaining n lines is one of the backup directives. You may assume that all input files are formatted correctly.

Sample input #1 (available as file “test2a.dat”):

```
10?01
6 2
10101
10001
10000
10010
11010
01010
```

Sample output #1:

```
Compatible backup directives: 1 2 3 4
```

Sample input #2 (available as file "test2b.dat"):

```
?0?01  
6 1  
10101  
01010  
10000  
10010  
11010  
10001
```

Sample output #2:

```
Compatible backup directives: 1 3 6
```

Sample input #3 (available as file "test2c.dat"):

```
10?11  
5 0  
10101  
01010  
10000  
10010  
11010
```

Sample output #3:

```
No compatible backup directives
```