reversed to “gub”. As a result, the output is “racgodgub”. If M > *n*, output the original string without inversing it.

Note: You must implement it by a stack; otherwise, no points will be given.

Test Case

Please test your program with Input, and then check the answers with Output.

Listing 19 : Palindrome

```
Input ：
10 ASDFASDFA
3 DFJIDOJBIICVJOZXOJQSA
Output ：
ASDFASDFA
JFDODIIBJVCIZOJJOXASQ
```

20. KMP (2)

Please read the given file (input.txt) that involves the NAME and INDEX data. At first, you should initiate the MAIN_STRING as ‘NULL’. You then will use MAIN_STRING to store the read data from the file (input.txt). The value of NAME is the data you should insert to the MAIN_STRING, and the INDEX data represents the index position where you should insert at the next time. For an instance, after the initiation, you read ‘jim’ and ‘1’ from file (input.txt), hence, the MAIN_STRING will be ‘jim’. Then, you keep reading ‘tom’ and ‘2’ from the file(input.txt), you should insert ‘tom’ into the ‘1’ position of the current MAIN_STRING, therefore, the MAIN_STRING will be ‘jtomim’. The last line “P:aaaaa” in the input file is used to implement the KMP algorithm.

After read file, implement the KMP algorithm to preprocess a read MAIN_STRING, and print the LSP[] which is used to skip characters while matching.

Listing 19 : Palindrome

```
Input ：
jim,1
tom,2
patty,3
helen,4
aaaaa,4
P:aaaaa
```

```
Output ：
Final string : jtopatheleaaaaantymim
Last string: jtopathelentymim
Lps = {0,1,2,3,4}
```