In [6]:

```
#1

def swap_case(s):
    list1=[]
    for i in s:
        if i.isalpha():
            if i.islower():
                list1.append(i.upper())
            elif i.isupper():
                list1.append(i.lower())
        else: list1.append(i)

    return ''.join(list1)
s = input()
result = swap_case(s)
print(result)
```

```
HackerRank.com presents "Pythonist 2".
hACKERrANK.COM PRESENTS "pYTHONIST 2".
```

In [7]:

```
#2

def split_and_join(line):
    # write your code here
    l=line.split(' ')
    l= '-'.join(l)
    return l

if __name__ == '__main__':
    line = input()
    result = split_and_join(line)
    print(result)
```

```
this is a string
this-is-a-string
```

In [8]:

```
#3

def print_full_name(first, last):

    return print('Hello {first} {last}! You just delved into python.'.format(first=first, last=last))

if __name__ == '__main__':
    first_name = input()
    last_name = input()
    print_full_name(first_name, last_name)
```

```
utkarsh
misha
Hello utkarsh misha! You just delved into python.
```

In [10]:

```
#4

def mutate_string(string, position, character):
    list1 = list(string)
    list1[position]=character

    return  ''.join(list1)




if __name__ == '__main__':
    s = input()
    i, c = input().split()
    s_new = mutate_string(s, int(i), c)
    print(s_new)
```

```
utkarshrathorenishamishra
5 n
utkarnhrathorenishamishra
```

In [11]:

```
#5

def count_substring(string, sub_string):
    x = [str(string.find(sub_string,i))  for i in range(0,len(string) )]
    a=list(set(x))
    a.remove('-1')
    return len(a)


if __name__ == '__main__':
    string = input().strip()
    sub_string = input().strip()

    count = count_substring(string, sub_string)
    print(count)
```

```
UTKARSHNISHA
UN
0
```

In [12]:

```
#6

s=str(input())
print(any([i for i in s if i.isalnum() ]))
print(any([i for i in s if i.isalpha() ]))
print(any([i for i in s if i.isdigit() ]))
print(any([i for i in s if i.islower() ]))
print(any([i for i in s if i.isupper() ]))
```

```
UTKARSHNISHA MISHRARATHORE
True
True
False
False
True
```

In [13]:

```
#7


thickness = int(input()) #This must be an odd number
c = 'H'

#Top Cone
for i in range(thickness):
    print((c*i).rjust(thickness-1)+c+(c*i).ljust(thickness-1))

#Top Pillars
for i in range(thickness+1):
    print((c*thickness).center(thickness*2)+(c*thickness).center(thickness*6))

#Middle Belt
for i in range((thickness+1)//2):
    print((c*thickness*5).center(thickness*6))

#Bottom Pillars
for i in range(thickness+1):
    print((c*thickness).center(thickness*2)+(c*thickness).center(thickness*6))

#Bottom Cone
for i in range(thickness):
    print(((c*(thickness-i-1)).rjust(thickness)+c+(c*(thickness-i-1)).ljust(thickness)).rjust(thickness*6))
```

```
5
    H
   HHH
  HHHHH
 HHHHHHH
HHHHHHHHH
  HHHHH               HHHHH
  HHHHH               HHHHH
  HHHHH               HHHHH
  HHHHH               HHHHH
  HHHHH               HHHHH
  HHHHH               HHHHH
  HHHHHHHHHHHHHHHHHHHHHHHHH
  HHHHHHHHHHHHHHHHHHHHHHHHH
  HHHHHHHHHHHHHHHHHHHHHHHHH
  HHHHH               HHHHH
  HHHHH               HHHHH
  HHHHH               HHHHH
  HHHHH               HHHHH
  HHHHH               HHHHH
  HHHHH               HHHHH
                    HHHHHHHHH
                     HHHHHHH
                      HHHHH
                       HHH
                        H
```

```
In [14]: #8

         import textwrap

         def wrap(string, max_width):
             return "\n".join([string[i:i+max_width] for i in range(0, len(string), max_width)])

         if __name__ == '__main__':
             string, max_width = input(), int(input())
             result = wrap(string, max_width)
             print(result)
```

```
ABCDEFGHIJKLIMNOQRSTUVWXYZ
4
ABCD
EFGH
IJKL
IMNO
QRST
UVWX
YZ
```

```
In [15]: #9

         n, m = map(int,input().split())
         pattern = [('.|.'*(2*i + 1)).center(m, '-') for i in range(n//2)]
         print('\n'.join(pattern + ['WELCOME'.center(m, '-')] + pattern[::-1]))
```

```
7 21
---------.|.---------
------.|..|..|.------
---.|..|..|..|..|.---
-------WELCOME-------
---.|..|..|..|..|.---
------.|..|..|.------
---------.|.---------
```

In [17]:

```
#10

def print_formatted(number):

    results = []

    for i in range(1, n+1):
        decimal = str(i)
        octal = str(oct(i)[2:])
        hex_ = str(hex(i)[2:]).upper()
        binary = str(bin(i)[2:])

        results.append([decimal, octal, hex_, binary])

    width = len(results[-1][3])  # largest binary number

    for i in results:
        print(*(rep.rjust(width) for rep in i))



if __name__ == '__main__':
    n = int(input())
    print_formatted(n)
```

```
50
     1      1      1      1
     2      2      2     10
     3      3      3     11
     4      4      4    100
     5      5      5    101
     6      6      6    110
     7      7      7    111
     8     10      8   1000
     9     11      9   1001
    10     12      A   1010
    11     13      B   1011
    12     14      C   1100
    13     15      D   1101
    14     16      E   1110
    15     17      F   1111
    16     20     10  10000
    17     21     11  10001
    18     22     12  10010
    19     23     13  10011
    20     24     14  10100
    21     25     15  10101
    22     26     16  10110
    23     27     17  10111
    24     30     18  11000
    25     31     19  11001
    26     32     1A  11010
    27     33     1B  11011
    28     34     1C  11100
    29     35     1D  11101
    30     36     1E  11110
    31     37     1F  11111
    32     40     20 100000
    33     41     21 100001
    34     42     22 100010
    35     43     23 100011
    36     44     24 100100
    37     45     25 100101
    38     46     26 100110
    39     47     27 100111
    40     50     28 101000
```

```
41     51     29 101001
42     52     2A 101010
43     53     2B 101011
44     54     2C 101100
45     55     2D 101101
46     56     2E 101110
47     57     2F 101111
48     60     30 110000
49     61     31 110001
50     62     32 110010
```

In [18]:

```
#11

def print_rangoli(size):
    from string import ascii_lowercase as chars
    heap = [(('-'.join(chars[i:n])[::-1]+'-'.join(chars[i:n])[1:])).center(4*n-3, '-') for i in range(n)]
    print(*(heap[::-1]+heap[1:]), sep="\n")

if __name__ == '__main__':
    n = int(input())
    print_rangoli(n)
```

```
15
----------------------------o----------------------------
--------------------------o-n-o--------------------------
------------------------o-n-m-n-o------------------------
----------------------o-n-m-l-m-n-o----------------------
--------------------o-n-m-l-k-l-m-n-o--------------------
------------------o-n-m-l-k-j-k-l-m-n-o------------------
----------------o-n-m-l-k-j-i-j-k-l-m-n-o----------------
--------------o-n-m-l-k-j-i-h-i-j-k-l-m-n-o--------------
------------o-n-m-l-k-j-i-h-g-h-i-j-k-l-m-n-o------------
----------o-n-m-l-k-j-i-h-g-f-g-h-i-j-k-l-m-n-o----------
--------o-n-m-l-k-j-i-h-g-f-e-f-g-h-i-j-k-l-m-n-o--------
------o-n-m-l-k-j-i-h-g-f-e-d-e-f-g-h-i-j-k-l-m-n-o------
----o-n-m-l-k-j-i-h-g-f-e-d-c-d-e-f-g-h-i-j-k-l-m-n-o----
--o-n-m-l-k-j-i-h-g-f-e-d-c-b-c-d-e-f-g-h-i-j-k-l-m-n-o--
o-n-m-l-k-j-i-h-g-f-e-d-c-b-a-b-c-d-e-f-g-h-i-j-k-l-m-n-o
--o-n-m-l-k-j-i-h-g-f-e-d-c-b-c-d-e-f-g-h-i-j-k-l-m-n-o--
----o-n-m-l-k-j-i-h-g-f-e-d-c-d-e-f-g-h-i-j-k-l-m-n-o----
------o-n-m-l-k-j-i-h-g-f-e-d-e-f-g-h-i-j-k-l-m-n-o------
--------o-n-m-l-k-j-i-h-g-f-e-f-g-h-i-j-k-l-m-n-o--------
----------o-n-m-l-k-j-i-h-g-f-g-h-i-j-k-l-m-n-o----------
------------o-n-m-l-k-j-i-h-g-h-i-j-k-l-m-n-o------------
--------------o-n-m-l-k-j-i-h-i-j-k-l-m-n-o--------------
----------------o-n-m-l-k-j-i-j-k-l-m-n-o----------------
------------------o-n-m-l-k-j-k-l-m-n-o------------------
--------------------o-n-m-l-k-l-m-n-o--------------------
----------------------o-n-m-l-m-n-o----------------------
------------------------o-n-m-n-o------------------------
--------------------------o-n-o--------------------------
----------------------------o----------------------------
```

In [ ]: