```
#
# Prints triangular stars for n lines
#    in an ascending manner
#
def ascending_stars(n):

n=int(raw_input("Enter the number : "))
ascending_stars(n)

#
#    Examples
#
#    1) Enter the number : 4
#
#    *
#    * *
#    * * *
#    * * * *
#
#    1) Enter the number : 5
#
#    *
#    * *
#    * * *
#    * * * *
#    * * * * *
#
```

```
#
# Prints triangular stars for n lines
#    in a descending manner
#
def descending_stars(n):

n=int(raw_input("Enter the number : "))
descending_stars(n)

#
#    Examples
#
#    1) Enter the number : 4
#
#    * * * *
#    * * *
#    * *
#    *
#
#    1) Enter the number : 5
#
#    * * * * *
#    * * * *
#    * * *
#    * *
#    *
#
```

```
# See the examples
def complete_stars(n):

n=int(raw_input("Enter the number : "))
complete_stars(n)

#
#    Examples
#
#    1) Enter the number : 4
#
#    *
#    * *
#    * * *
#    * * * *
#    * * *
#    * *
#    *
#
#
#    1) Enter the number : 5
#
#    *
#    * *
#    * * *
#    * * * *
#    * * * * *
#    * * * *
#    * * *
#    * *
#    *
#
```

```
# Counting

# copy the code for the first three functions
def ascending_stars(n):

def descending_stars(n):

def complete_stars(n):

# write the code that computes the number of stars for each shape
#     See the example for precise description
def count_ascending_stars(n):

def count_descending_stars(n):

def count_complete_stars(n):


n=int(raw_input("Enter the number : "))
ascending_stars(n)
print "Number of stars in above shape = " count_ascending_stars(n)
descending_stars(n)
print "Number of stars in above shape = " count_descending_stars(n)
complete_stars(n)
print "Number of stars in above shape = " count_complete_stars(n)

#    1) Enter the number : 5
#
#    *
#    * *
#    * * *
#    * * * *
#    * * * * *
#
#    Number of stars in above shape = 15
#
#    * * * * *
#    * * * *
#    * * *
#    * *
#    *
#
#    Number of stars in above shape = 15
#
#    *
#    * *
#    * * *
#    * * * *
#    * * * * *
#    * * * *
#    * * *
#    * *
#    *
#
#    Number of stars in above shape = 25
#
```

```
# exactly_once(str,ch) returns true if the character ch occurs exactly once in str
#        otherwise return False
def exactly_once(str,ch):


str=raw_input("Enter the string : ")
ch=raw_input("Enter the character : ")
print exactly_once(str,ch)

#
#    Example
#
#    Enter the string : ubuntu
#    Enter the character : b
#    True
#
#    Enter the string : apple
#    Enter the character : p
#    False
#
#    Enter the string : windows
#    Enter the character : w
#    False
```

```
# Check whether a string is a palindrome or not in at least two different ways
def pal1(str):

def pal2(str):

str = raw_input("Enter the string : ")
print pal1(str)
print pal2(str)

#
#    Examples
#
#    1) Enter the string : able was i ere i saw elba
#       True
#       True
#
```

```
#
# Delete the first occurence of a letter from the string
#     with and without a function call
#

def delete_first_occurrence(s,ch):

str1 = raw_input("Enter the string : ")
ch = raw_input("Enter the character : ")
str2=delete_first_occurrence(str1,ch)
#
# You can write more code here to modify str1
#
print "The string after deletion is ",str1
print "The string after deletion is ",str2


#
#    Examples
#
#    1) Enter the string : clarify
#        Enter the character : a
#        The string after deletion is clrify
#        The string after deletion is clrify
#
```