Write both an iterative and recursive function called `is_even()` that determines if a given number is even or not.

```python
def is_even(num):
    return num%2 == 0

def recursive_is_even(num):
    if n == 0:
    return True
    elif n == 1:
    return False
    else:
    return recursive_is_even(num-2)
```

```
[nathan@archlinux ~]$ python -i main.py
>>> recursive_is_even(69)
False
>>>
```

Write both an iterative and recursive function called `letter_frequency()` that determines how many times a given letter occurs in a provided string.

```python
def letter_freq(string,letter):
    return string.count(letter)

def recursive_letter_freq(string, letter):
    if len(string) == 0:
    return 0
    elif string[0] == letter:
    return 1 + recursive_letter_freq(string[1:], letter)
    else:
    return recursive_letter_freq(string[1:], letter)
```

```
>>> recursive_letter_freq("ugabuga","a")
2
>>>
```

Write both an iterative and recursive function called `reverse()` that displays a provided string backwards.

```python
def reverse(string):
    return string[::-1]

def reverse_recursive(string):
    if len(string) == 0:
    return ""
    else:
    return string[-1] + reverse_recursive(string[:-1])
```

```
>>> reverse_recursive("racecar")
'racecar'
>>> reverse_recursive("hah! Goteem!")
'!meetoG !hah'
>>>
```

Write both an iterative and recursive function called `is_prime()` that determines if a given number is a prime number or not.

```
def is_prime(num):
    for i in range(2,int((num**1/2)+1)):
    if num % i == 0:
        return False
    return True

def is_prime_recursive(num,i):
    if i == int(num**1/2 + 1):
    return True
    if num % i == 0:
    return False
    else:
    return is_prime_recursive(num,i+1)
```

```
[nathan@archlinux ~]$ python -i main.py
>>> is_prime_recursive(69)
False
>>>
```

Write both an iterative and recursive function called `binary()` that converts decimal numbers to binary/base 2.

```
def decimal_to_binary(num):
    binary = []
    while num != 0:
    num,remainder = divmod(num,2);
    binary.append(remainder)
    print(''.join(str(x) for x in binary)[::-1])


def rec_decimal_to_binary(n):
    if n == 0:
    return "0"
    else:
    return rec_decimal_to_binary(n // 2) + str(n % 2)
```

```
>>> rec_decimal_to_binary(10)
'01010'
>>>
```

Write both an iterative and recursive function called `fib()` that calculates the Fibonacci number for any positive integer, defined as follows:

```python
def fib_iterative(n):
    if n == 1:
    return 0
    elif n == 2:
    return 1
    else:
    a, b = 0, 1
    for _ in range(2, n):
        a, b = b, a + b
    return b

def fib_recursive(n):
    if n == 1:
    return 0
    elif n == 2:
    return 1
    else:
    return fib_recursive(n-1) + fib_recursive(n-2)
```

```
[nathan@archlinux ~]$ python -i main.py
>>> fib_
fib_iterative(  fib_recursive(
>>> fib_recursive(5)
3
>>> fib_recursive(10)
34
>>>
```

Write both an iterative and recursive function called `sumlist()` to calculate the sum of a list of numbers.

```python
def sum_iterative(num):
    total = 0
    for number in num:
    total += number
    return total

def sum_recursive(arr):
    if len(arr) == 0:
    return 0
    return arr[0] + sum_recursive(arr[1:])
```

```
>>> fib_recursive(10)
34
>>> sum_recursive([1,2,3,4,5])
15
>>>
```

Write a recursive function called `harmonic()` to calculate the harmonic sum of `n-1`.

```
def harmonic_recursive(n):
    if n == 0:
    return 0
    return (1/n) + harmonic_recursive(n-1)
```

```
[nathan@archlinux ~]$ python -i main.py
>>> harmonic_recursive(20)
3.597739657143682
```

Write a recursive function `power()` to calculate the value of `a` to the power `b`.

```
def power_recursive(a,b):
    if b == 0:
    return 1
    return a*power_recursive(a,b-1)
```

```
>>> power_recursive(2,3)
8
```

Write a recursive function `greatest_common_denominator()` to find the greatest common divisor (GCD) of two integers.

```
def greatest_common_denominator(a,b):
    remainder = a % b
    if remainder == 0:
    return b
    a,b = b,remainder
    return greatest_common_denominator(a,b)
```

```
>>> greatest_common_denominator(69,420)
3
```

Write both an iterative and recursive function that displays the rows of asterisks given below,

****

****

****

******

```
****

**
```

```
def pattern_01_iter():
    rows = 6
    for i in range(rows, 0, -1):
    print(" " * (rows - i) + "*" * (i * 2))

def pattern_01_rec(i=6):
    if i == 0:
    return
    rows=6
    print(" "*(rows-i)+"*"*(i*2))
    i-=1
    pattern_01_rec(i)
```

```
>>> pattern_01_rec()
************
 **********
  ********
   ******
    ****
     **
```

Write both an iterative and recursive function that displays the rows of asterisks given below,

```
**

****

******

**********

************

**************
```

```
def pattern_02_iter():
    rows = 7
    for i in range(0, rows, 1):
    print(" " * (rows - i) + "*" * (i * 2))

def pattern_02_rec(i=1):
    rows = 6
    print(" " * (rows - i) + "*" * (i * 2))
    if i == rows:
    return
    i+=1
    pattern_02_rec(i)
```

```
      **
>>> pattern_02_rec()
      **
     ****
    ******
   ********
  **********
************
```

Write a Python program that asks the user to think of a number between 1 and 100 and then attempt to guess what that number is by iteratively(non-recursive) showing the user a number guess and then asking the user if the guess is correct, or higher or lower than the user's number.
Write a recursive version of the program in (**1**)

```
import random
def guessing_game():
     lower = 1
     upper = 100
     while True:
     guess = random.randint(lower, upper)
     guessed = input(f"Is your number {guess}? (y/n)")
     if guessed == "y":
          print(":P im tottally a mind reader")
          break
     user_input = input(f"Was the number {guess} higher or lower than your number?
(h/l)")
     if user_input == "h":
          upper = guess - 1
     else:
          lower = guess + 1


def guessing_game_recur(lower=1,upper=100):
     guess = random.randint(lower,upper)
     guessed = input(f"Is your number {guess}? (y/n)")
     if guessed == "y":
     print(":P im tottally a mind reader")
     return
     user_input = input(f"Was the number {guess} higher or lower than your number?
(h/l)")
     if user_input == "h":
     guessing_game_recur(lower,(guess-1))
     else:
     lower = guess + 1
     guessing_game_recur((guess+1),upper)
```

```
>>> guessing_game_recur()
Is your number 16? (y/n)n
Was the number 16 higher or lower than your number? (h/l)l
Is your number 67? (y/n)n
Was the number 67 higher or lower than your number? (h/l)h
Is your number 42? (y/n)n
Was the number 42 higher or lower than your number? (h/l)l
Is your number 58? (y/n)n
Was the number 58 higher or lower than your number? (h/l)h
Is your number 54? (y/n)n
Was the number 54 higher or lower than your number? (h/l)h
Is your number 50? (y/n)n
Was the number 50 higher or lower than your number? (h/l)l
Is your number 52? (y/n)n
Was the number 52 higher or lower than your number? (h/l)l
Is your number 53? (y/n)y
:P im tottally a mind reader
```