Module02_Day08_Recursion_1

December 16, 2022

Recusrsion 1

- Solving a problem using smaller instance of same problem
  - Function calling itself

Sum of first n natural numbers

```
# sum(3) -> 1 + 2 + 3
# sum(4) -> sum(3) + 4
# sum(n) -> sum(n-1) + n
```

[ ]:
```
def add(n):
    if n == 1:
        return 1
    return add(n-1) + n
```

[ ]:
```
add(5)
```

[ ]:
```
15
```

[ ]:
```
def factorial(n):
    if n == 1:
        return 1
    return factorial(n-1) * n
```

[ ]:
```
def factorial2(n):
    if n == 0:
        return 1
    return factorial2(n-1) * n
```

[ ]:
```
factorial2(3)
```

[ ]:
```
6
```

[ ]:
```
factorial(0)
```

[ ]:
```
1
```

[ ]:
```
def fibonacci(n):
    if n <= 0:
        return n + 1
    return fibonacci(n-1) + fibonacci(n-2)
```

[ ]:
```
fibonacci(4)
```

[ ]: 5

[ ]:
```
def power2(n):
    if n == 1:
        return 2
    return power2(n-1) * 2
```

[ ]:
```
power2(3)
```

[ ]: 8

[ ]:
```
def ispower(n,a):
    if n == 1:
        return True
    if n % a !=0 :
        return False
    return ispower(n/a,a)
```

[ ]:
```
ispower(9,3)
```

[ ]: True

[ ]:
```
def length(str1 ,i = 0):
    if str1[i:] == "":
        return i
    return length(str1, i + 1)
```

[ ]:
```
length("hell")
```

[ ]: 4

[ ]:
```
def length2(str1):
    if str1 == "":
        return 0
    return 1 + length2(str1[1:])
```

[ ]:
```
length2("hell")
```

[ ]: 4

[ ]:
```
def print_num(n):
    if n == 0:
        return
    print(n)
    print_num(n-1)
```

[ ]:
```
print_num(5)
```

```
5
4
3
2
1
```

[ ]:
```
def pow(a,n):
    if n <= 1:
        return a
    return pow(a,n-1) * a
```

[ ]:
```
pow(3,2)
```

[ ]:
```
9
```

[ ]:
```
def opt_pow(a,n):
    if n <= 1:
        return a
    if n  % 2 == 0:
        temp = opt_pow(a,n/2)
        return temp * temp
    else:
        temp = opt_pow(a,n//2)
        return temp * temp * a
```

[ ]:
```
opt_pow(3,3)
```

[ ]:
```
27
```

[ ]:
```
def print_no(n):
    if n == 1:
        print(n)
        return
    print_no(n-1)
    print(n)
```

[ ]:
```
print_no(5)
```

```
1
2
3
```

```
4
5
```

[ ]:
```
def fun(i,j):
      if(i==0):
         return j
      else:
         return fun(i-1,j+1)

fun(4,8)
```

[ ]:
```
12
```

[ ]:
```
def count_freq(x, y):
  if(y!=1):
    if(x!=1):
      print("*")
      count_freq(x/2,y)
    else:
      print("#")
      y=y-1
      count_freq(x,y)

count_freq(256,4)
```

```
*
*
*
*
*
*
*
*
*
#
#
#
```

[ ]:
```
def fun(i,j):
  if(i==0):
    return j
  else:
    return fun(j,i-5)

print(fun(15,8))
```

```
-7
```

[ ]:
```
def countHi(s, count=0):
    if len(s) < 1:
        return count
    if s[:2] == "hi":
        return countHi(s[1:],count+1)
    else:
        return countHi(s[1:],count)

countHi("hishisha")
```

[ ]: 2