




2-8

(Bubble Sort Algorithm)

```
array=[ 5, 3, 4, 6, 9, 1, 7, 2, 0, 8]
def BubbleSort(data):
    n = len(data)
    for i in range(n - 1):
        for j in range(n - 1 - i):
            if data[j] > data[j+1]:
                temp = data[j]
                data[j] = data[j+1]
                data[j+1] = temp
    return data


print(BubbleSort(array))
```



```
lower = 100
upper = 200
for num in range(lower, upper+1):
    if num > 1:
        for i in range(2, num):
            if num % i == 0:
                break
        else: #for else else
            print(num)
```



(Least Common Mutiple, LCM)

```
def LCM(num1, num2):
    if num1 > num2: #
        greater = num1
```

```

else:
    greater = num2
while True:
    if (greater % num1 == 0 ) and (greater % num2 == 0):    #[]
        break
    greater += 1
return greater

n1 = int(input(" []"))
n2 = int(input(" []"))

print(" [] = " , LCM(n1, n2))

```

[]

(Greatest Common Divisor,GCD) [] (Highest Common Factor, HCF)

```

def GCD(num1, num2):
    if num1 > num2 :
        smaller = num2
    else:
        smaller = num1
    for i in range(1, smaller + 1):
        if (num1 % i == 0) and (num2 % i == 0):    #i[]
            factor = i
    return factor

n1 = int(input(" []"))
n2 = int(input(" []"))

print(" [] = " , GCD(n1, n2))

```

[]

```

n = int(input(" []"))
def rev_num(num):
    rev = 0

```

```
while num > 0:
    reminder = num % 10
    rev = (rev * 10) + reminder
    num = num // 10
return rev

print("反转后的数字=", rev_num(n))
```



```
s = input("请输入字符串: ")

def rev_str(string):
    x = ""
    for i in string:
        x = i + x
    return x

print(rev_str(s))
```

Revision #4

Created 19 November 2022 16:23:27 by CSG

Updated 10 December 2022 15:13:00 by CSG