Algorithms

- min max in an list

In [1]:

```
lst1 = [int(item) for item in input("Enter the list items : ").split()]
```

In [2]:

```
listt = [ int(i)  for i in input("list").split()]
print(listt)
```

```
[1]
```

In [18]:

```
# min   in an list
listt = [ int(i)  for i in input("list").split()]
print(listt)
min =46547578896567
for i in listt:
    if min> i :
        min = i
print(min)
```

```
1
```

In [19]:

```
#linear Search
```

In [3]:

```
listt = [ i  for i in range(1000)]

x = 0
b = int(input())
for i in listt:
    if i==b:
        x = 1
        print("found ")
        break

if(x==0):
    print("Not Found")
```

```
found
```

- binary Search --> Sorted Data [ ]

In [5]:

```
listt = [ i  for i in range(1000)]

start = 0
x = 0
no = int(input())
end = len(listt)-1
count = 0
while(start <=end):
    mid = (start + end) // 2
    if(listt[mid]== no):
        print("found")
        x=1
        break
    elif(listt[mid] > no):
        end= mid - 1
    else:
        start = mid + 1
    count+=1
if(x==0):
    print("Not Found")
print(count)
```

```
found
8
```

In [13]:

```
listt =[]
for i in input("Enter the no ").split():
    listt.append(int(i))
print(listt)
```

```
[1]
```

In [ ]:

```

```

In [10]:

```
listt= []
for i in input("Enter the no ").split():
    listt.append(int(i))
print(listt)
```

```
[1, 2, 4, 55, 6, 6, 644]
```

- sorting

In [4]:

```
listt =  [67, 45, 40, 34, 22, 17, 11]
i = 0
n = len(listt)

while(i<n):
    j=0
    while(j <n- i -1):
        if(listt[j]>listt[j+1]):
            listt[j], listt[j+1] = listt[j+1], listt[j]
        j+=1
    i+=1
print(listt)
```

```
[11, 17, 22, 34, 40, 45, 67]
```

function :-function is a block of of organised ,reuseable, code ,that is used to perform a single related action function provide better modularity for your application and heigh degree of code reusable.

In [13]:

```
### Sorting and use swap function
def swap(i , j , l ):
    l[i] , l[j] = l[j] , l[i]
```

In [14]:

```
listt =  [67, 45, 40, 34, 22, 17, 11]
i = 0
n = len(listt)
while(i<n):
    j=0
    while(j <n- i -1):
        if(listt[j]>listt[j+1]):
            swap(j ,j+1 , listt)
        j+=1
    i+=1
print(listt)
```

```
[11, 17, 22, 34, 40, 45, 67]
```

In [23]:

```
### WAP to swap 1st and last value of the list
listt =  [67, 45, 40, 34, 22, 17, 11]
def swap(i , j, l ):
    l[i] , l[j] = l[j] , l[i]
swap(0 , len(listt)-1  , listt)
```

In [53]:

```
import random
a = random.random()
b = random.randrange(20, 50)
```

In [54]:

```
print(b)
```

```
49
```

In [34]:

```

```

Out[34]:

```
0.462245319162521
```

In [41]:

```
a = input()
b=input()
print("before function ")
print(count(a , b ))
print("after function ")

def count(a ,b):
    c = 0
    for i in a:
        if i==b:
            c+=1
    print("in function ")
    return c
```

```
before function
in function
2
after function
```

- WAP to check if a str is palindrome or (not using function)

In [57]:

```
string = input()
def palidrome(string ):
    i = 0
    j = len(string) -1

    while(i <=j):
        if(string[i]!=string[j]):
            return False
        i+=1
        j-=1
    return True
if(palidrome(string)):
    print("Palindrome")
else:
    print("Not Palindrome")
```

```
Not Palindrome
```

In [ ]:

```
[1 , 3, , 9 ]
[1 , 4 ,0 , 0 ]
```