

daa-pertemuan-4

September 24, 2023

1 Pertemuan 4

2 Algoritma Swap

```
[27]: var1 = 2
      var2 = 3
      print(var1, var2)
      var1, var2 = var2, var1
      print(var1, var2)
```

```
2 3
3 2
```

3 Latihan

```
[28]: var1 = 2
      var2 = 3
      var3 = 4
      print(var1, var2, var3)
      var1, var2, var3 = var3, var1, var2
      print(var1, var2, var3)
```

```
2 3 4
4 2 3
```

4 Bubble Sort

```
[29]: list = [34, 23, 12, 45, 9, 1, 24]
      lastElementIndex = len(list)-1
      print(0, list)
      for idx in range(lastElementIndex):
          if list[idx] > list[idx+1]:
              list[idx], list[idx+1] = list[idx+1], list[idx]
          print(idx+1, list)
```

```
0 [34, 23, 12, 45, 9, 1, 24]
1 [23, 34, 12, 45, 9, 1, 24]
```

```

2 [23, 12, 34, 45, 9, 1, 24]
3 [23, 12, 34, 45, 9, 1, 24]
4 [23, 12, 34, 9, 45, 1, 24]
5 [23, 12, 34, 9, 1, 45, 24]
6 [23, 12, 34, 9, 1, 24, 45]

```

```
[30]: # Lets make this nonstop until there is no more swap
```

```

def BubbleSort(list):
    lastElementIndex = len(list) - 1
    for passno in range(lastElementIndex, 0, -1):
        for indx in range(passno):
            if list[indx] > list[indx + 1]:
                list[indx], list[indx + 1] = list[indx + 1], list[indx]
            print("passno = ", passno, 'Index = ', indx, "list = ", list)
    return list

```

```
[31]: list = BubbleSort([3,5,7,6,9,50,10,13,12,2])
print(list)
```

```

passno = 9 Index = 8 list = [3, 5, 6, 7, 9, 10, 13, 12, 2, 50]
passno = 8 Index = 7 list = [3, 5, 6, 7, 9, 10, 12, 2, 13, 50]
passno = 7 Index = 6 list = [3, 5, 6, 7, 9, 10, 2, 12, 13, 50]
passno = 6 Index = 5 list = [3, 5, 6, 7, 9, 2, 10, 12, 13, 50]
passno = 5 Index = 4 list = [3, 5, 6, 7, 2, 9, 10, 12, 13, 50]
passno = 4 Index = 3 list = [3, 5, 6, 2, 7, 9, 10, 12, 13, 50]
passno = 3 Index = 2 list = [3, 5, 2, 6, 7, 9, 10, 12, 13, 50]
passno = 2 Index = 1 list = [3, 2, 5, 6, 7, 9, 10, 12, 13, 50]
passno = 1 Index = 0 list = [2, 3, 5, 6, 7, 9, 10, 12, 13, 50]
[2, 3, 5, 6, 7, 9, 10, 12, 13, 50]

```

5 Latihan

```

[32]: def Latihan_BubbleSort(urutan):
    panjang_list = len(urutan) - 1
    for i in range(panjang_list, 0, -1):
        for index in range(i):
            if urutan[index] > urutan[index + 1]:
                urutan[index], urutan[index+1] = urutan[index + 1],
urutan[index]
            print("i = ", i, "index = ", index, "list = ", urutan)
    return urutan

```

```
[33]: print(Latihan_BubbleSort([100,20,60,90,40,30,10]))
```

```

i = 6 index = 5 list = [20, 60, 90, 40, 30, 10, 100]
i = 5 index = 4 list = [20, 60, 40, 30, 10, 90, 100]

```

```

i = 4 index = 3 list = [20, 40, 30, 10, 60, 90, 100]
i = 3 index = 2 list = [20, 30, 10, 40, 60, 90, 100]
i = 2 index = 1 list = [20, 10, 30, 40, 60, 90, 100]
i = 1 index = 0 list = [10, 20, 30, 40, 60, 90, 100]
[10, 20, 30, 40, 60, 90, 100]

```

6 Insertion Sort

```

[34]: def Insertion_sort (list):
        for i in range(1, len(list)):
            j = i - 1
            next = list[i]
            print("Proses pada i = ", i, " Dengan Nilai banding = " , next)
            while (list[j] > next) and (j >= 0):
                list[j + 1] = list[j]
                print("j = ", j, "list = ", list)
                j = j - 1
            list[j+1] = next
            print("i = ",i, "list = ",list)
        return list

print(Insertion_sort([25,26,22,24,27,23,21]))

```

```

Proses pada i = 1 Dengan Nilai banding = 26
i = 1 list = [25, 26, 22, 24, 27, 23, 21]
Proses pada i = 2 Dengan Nilai banding = 22
j = 1 list = [25, 26, 26, 24, 27, 23, 21]
j = 0 list = [25, 25, 26, 24, 27, 23, 21]
i = 2 list = [22, 25, 26, 24, 27, 23, 21]
Proses pada i = 3 Dengan Nilai banding = 24
j = 2 list = [22, 25, 26, 26, 27, 23, 21]
j = 1 list = [22, 25, 25, 26, 27, 23, 21]
i = 3 list = [22, 24, 25, 26, 27, 23, 21]
Proses pada i = 4 Dengan Nilai banding = 27
i = 4 list = [22, 24, 25, 26, 27, 23, 21]
Proses pada i = 5 Dengan Nilai banding = 23
j = 4 list = [22, 24, 25, 26, 27, 27, 21]
j = 3 list = [22, 24, 25, 26, 26, 27, 21]
j = 2 list = [22, 24, 25, 25, 26, 27, 21]
j = 1 list = [22, 24, 24, 25, 26, 27, 21]
i = 5 list = [22, 23, 24, 25, 26, 27, 21]
Proses pada i = 6 Dengan Nilai banding = 21
j = 5 list = [22, 23, 24, 25, 26, 27, 27]
j = 4 list = [22, 23, 24, 25, 26, 26, 27]
j = 3 list = [22, 23, 24, 25, 25, 26, 27]
j = 2 list = [22, 23, 24, 24, 25, 26, 27]
j = 1 list = [22, 23, 23, 24, 25, 26, 27]

```

```
j = 0 list = [22, 22, 23, 24, 25, 26, 27]
i = 6 list = [21, 22, 23, 24, 25, 26, 27]
[21, 22, 23, 24, 25, 26, 27]
```

7 Penjelasan Insertion Sort

Insertion sort merupakan salah satu algoritma yang dapat digunakan untuk melakukan sorting suatu angka. Prinsip kerja dari sort ini adalah angka yang di urutkan akan di bandingkan dengan angka yang ada di sebelah kirinya. Jika angka di sebelah kirinya lebih besar maka angka tersebut akan di tukar posisinya dengan angka yang di sebelah kirinya. Proses ini akan terus berlanjut sampai angka yang di sebelah kirinya lebih kecil dari angka yang sedang di urutkan. Jika di dalam bubble sort terdapat istilah swap maka di insertion sort terdapat istilah shifting. Shifting merupakan proses menggeser angka ke kanan untuk menyediakan ruang bagi angka yang akan di urutkan.

8 Latihan

```
[35]: def metode_sorting_inverction(list):
        for i in range(1 , len(list)):
            j = i - 1
            next = list[i]
            while(list[j] > next) and (j >=0):
                list[j + 1] = list[j]
                # if u use bubble sort you must save the next in here like this
                # list[j] = next
                # and you compare list [ j -1 ] with list [j]
                j = j -1
            list[j + 1] = next # Why j + 1 karena j di loop hingga -1
        return list
```

```
[36]: print(metode_sorting_inverction([89,12,57,16,25,11,75]))
```

```
[11, 12, 16, 25, 57, 75, 89]
```

9 Selection Sort

```
[37]: def Selection_sort(list):
        for i in range(len(list) - 1, 0, -1):
            angka_ingin_banding = 0
            for j in range(0, i + 1):
                if (list[angka_ingin_banding] < list[j]):
                    angka_ingin_banding = j
            list[i], list[angka_ingin_banding] = list[angka_ingin_banding], list[i]
        return list
```

```
[38]: print(Selection_sort([70,15,25,19,34,44]))
```

[15, 19, 25, 34, 44, 70]

10 Linear Search

```
[39]: def search(Data, target):  
  
    found = False  
    for i in range(len(Data)):  
        if Data[i] == target:  
            found = True  
            print("Data ditemukan pada index ke - ", i)  
            break  
    if found == False:  
        print("Data tidak ditemukan")
```

```
[40]: list = [12, 33, 11, 99, 22, 55, 90]  
search(list, 55)  
search(list, 91)
```

Data ditemukan pada index ke - 5
Data tidak ditemukan

```
[41]: def linear_search(list, target):  
    Index = 0  
    found = False  
    penanda = 0  
    while(Index < len(list)) and found is False:  
        if list[Index] == target:  
            found = True  
            penanda = Index  
        else:  
            Index = Index + 1  
    return found
```

```
[42]: list = [12, 33, 11, 99, 22, 55, 90]  
print(linear_search(list, 12))  
print(linear_search(list, 91))
```

True
False

11 Latihan

```
[43]: def linearSearch(ListData, key):  
    list = []  
    for i in ListData:  
        list.append(i)
```

```

index = 0
found = False
huruf = 0
while index < len(list) and found is False:
    if list[index] == key:
        found = True
        huruf = index
    else:
        index = index + 1
return found, huruf

```

```

[44]: line = "yuiwoaquip"
data = linearSearch(line, 'a')
print(data)

```

(True, 5)

12 Binary Search

```

[45]: def BinarySearch(list, item):
    first = 0
    last = len(list) - 1
    found = False
    while first <= last and not found:
        midpoint = (first + last) // 2
        if list[midpoint] == item:
            found = True
        else:
            if item < list[midpoint]:
                last = midpoint - 1
            else:
                first = midpoint + 1
    return found

```

```

[46]: list = [12,33,11,99,22,55,90]
sorted_list = Selection_sort(list)
print(sorted_list)
print(BinarySearch(sorted_list, 12))

```

[11, 12, 22, 33, 55, 90, 99]
True

13 Latihan

```
[47]: def binary_search (data, target):  
    list = []  
    index_target = 0  
    for i in data:  
        list.append(i)  
    print(list)  
    for i in range (len(list)):  
        if list[i] == target:  
            index_target = i  
    print("Index target = ", index_target)  
  
    first = 0  
    last = len(list) - 1  
    found = False  
    while first <= last and not found:  
        mid_point = (first + last) // 2  
        if list[mid_point] == list[index_target]:  
            nilai = list[mid_point]  
            found = True  
        elif mid_point > index_target:  
            last = mid_point - 1  
            print("last = ", last)  
        else:  
            first = mid_point + 1  
            print("First = ", first)  
    return found, nilai
```

```
[48]: print(binary_search("yuiwoaquip", 'a'))
```

```
['y', 'u', 'i', 'w', 'o', 'a', 'q', 'u', 'i', 'p']  
Index target = 5  
First = 5  
last = 6  
(True, 'a')
```

14 Interpolation Search

Kalo tadi binary search itu kan berdasarkan nilai tengah pas nyarinya. Kalo interpolation search ini menggunakan nilai target untuk memperkirakan posisi elemen dalam array yang diurutkan. Contoh kalo kalian mau nyari river. Kita akan menggunakan informasi ini untuk interpolasi dan mulai mencari kata - kata yang dimulai dengan 'r'.

```
[49]: def IntPolsearch(list, x):  
    idx0 = 0  
    idxn = (len(list) - 1)
```

```

found = False
while idx0 <= idxn and x >= list[idx0] and x <= list[idxn]:
    # Find the mid point
    mid = idx0 + int(((float(idxn - idx0) / (list[idxn] - list[idx0])) * (x -
↪ list[idx0])))
    #Compare the value at mid point with search value
    if list[mid] == x:
        found = True
        return found
    if list[mid] < x:
        idx0 = mid + 1
return found

```

```

[50]: list = [12,33,11,99,22,55,90]
sorted_list = Selection_sort(list)
print(sorted_list)
print(IntPolsearch(sorted_list, 12))
print(IntPolsearch(sorted_list, 91))
print(IntPolsearch(sorted_list, 99))

```

[11, 12, 22, 33, 55, 90, 99]

True

False

True

15 Latihan

```

[51]: def interpolation_sort (list, target):
    kata = []
    index_huruf = []
    index_target = -1
    Found = False
    for i in list:
        kata.append(i)
    for i in range(len(kata)):
        if kata[i] == target:
            index_target = i + 1
            index_huruf.append(i + 1)
    print(kata)
    print(index_huruf)
    idx0 = 0
    idxn = len(kata) - 1
    print(idxn)
    while (idx0 <= idxn) and (index_target >= 0):
        mid = idx0 + int(((float((idxn - idx0) / (index_huruf[idxn] -
↪ index_huruf[idx0])) * (index_target - index_huruf[idxn])))

```



```

    print("Mid = ", mid, "indx0 = ", indx0)
    if kata[mid - 1] == target:
        Found = True
        return Found
    if mid < index_target:
        indx0 = mid + 1
    return Found

```

```

[52]: kata = "yuiwoaqujp"
      print(interpolation_sort(kata, "u"))

```

```

['y', 'u', 'i', 'w', 'o', 'a', 'q', 'u', 'j', 'p']
[1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
9
Mid = -2 indx0 = 0
True

```