

PDP-12

Array dan Sorting

Petunjuk :

Dalam modul ini , mirip dengan PDP-11a, perbedaanya masalah yang harus diselesaikan adalah harga ekstrim dan pengurutan data. Standar file yang digunakan ada 2, **testArraySort.py** dan **arraysort.py**

```
#Berikut driver untuk menguji solusi anda
#file : testArraySort.py
# -*- coding: utf-8 -*-
from arraysort import *

data1=[2,42,32,12,22,39,15,8,4,20]

Function main():
    TulisTabelData(data1)
    print("Cari 22 dalam data    : {}".format(CariElm(data1,22)))
    print("Cari idx elemen 22    : {}".format(CariIdx(data1,22)))
    print("Elemen Max data1 : {}".format(getValueMax(data1)))
    print("Index  Max data1 : {}".format(getIndexMax(data1)))
    sorted1=countingSort(data1,0,9999)
    TulisTabelData(sorted1)
    sorted2=selectionSort(data1)
    TulisTabelData(sorted2)
    sorted3=insertionSort(data1)
    TulisTabelData(sorted3)
    sorted4=counting_sort_mm(data1,0,9999)
    TulisTabelData(sorted4)

    print("Ta da ! ")

if __name__ == '__main__':
    main()
```

```
#file : arraysort.py
# -*- coding: utf-8 -*-
from math import *

Function panjang(data)
# mirip dengan PDP-11a

Function BacaTabelInt()
# mirip dengan PDP-11a
```

```

Function TulisTabelData(data)
# mirip dengan PDP-11a

Function CariElm(data,elm)
# mirip dengan PDP-11a

Function CariIdx(data,elm)
# mirip dengan PDP-11a

Function getValueMax(data) → Integer
#mengembalikan nilai elemen terbesar dari data
  maxi ← -9999
  i Iterate data
    jika maxi<i maka
      maxi ← i
  → maxi

Function getIndexMax(data)→ Integer
#mengembalikan posisi index elemen terbesar dari data
  maxi ← -9999;idx ← -1
  i Iterate range(panjang(data))
    jika maxi< data[i] maka
      maxi ← data[i]
      idx ← i
  → idx

Function countingSort(data, min, max)→ data
#buat list/array isi 0 sebesar max-min+1, untuk menyimpan
frekuensi
  cnt ← [0] * (max - min + 1)
#traversal pd data dan simpan frekuensi di cnt
  x Iterate data
    cntx - min = cntx - min +1
# buat list/array baru berdasarkan frekuensi
  sorted ← []
  x,n Iterate enumerate(cnt,min) # lihat cheat sheet
    i Iterate range(n)
      sorted ← sorted + [x]
  → sorted

Function counting_sort_mm(data,mini,maxi) → data
  p ← panjang(data)
  size ← maxi-mini+1
  count ← [0]*size
  i Iterate range(0,p)
    countdatai - mini ← countdatai - mini + 1

```

```

i Iterate range(mini, maxi)
  z ← 0
  j Iterate range(0, counti-mini-1):
    dataz ← i
    z ← z + 1
→ data

```

```

Function selectionSort(data) → data
p ← panjang(data)
i Iterate range(0,p)
  m ← i
  j Iterate range(i,p)
    Jika dataj < datam maka
      m ← j
  t ← datai
  datai ← datam
  datam ← t
→ data

```

```

Function insertionSort(data) →
p ← panjang(data)
i Iterate range (1,p)
  tmp ← data[i]
  j ← i
  Selama (j > 0 dan tmp < dataj-1) maka
    dataj ← dataj-1
    j ← j - 1
  dataj ← tmp
→ data

```