

Modul PDP-08

TITIK

Petunjuk :

Dalam Tugas ini anda harus menyelesaikan dengan standar 2 file (**testTitik.py**, **titik.py**). Masalah yang harus anda selesaikan adalah mengenai pengolahan suatu titik dalam koordinat **kartesian** (seperti dalam matematika smp/sma).

Berikut contoh output yang di inginkan untuk file **testTitik.py**

Buat dan Tulis point P1,P2,P3 dan P4

P1: (2.00,2.00)

P2: (-2.00,2.00)

P3: (2.00,-2.00)

P4: (-2.00,-2.00)

```
GetAbsis   P1      : 2
GetOrdinat P1      : 2
P4 stlh diset 4,4  : (4.00,4.00)
P5 stlh diset -2,2 : (-2.00,2.00)
Tulis P5 =P1+P2    : (2.00,6.00)
Tulis P5 =P1-P2    : (6.00,2.00)
Tulis P5 =P1.P2    : (-8.00,8.00)
Equal      P1 P2   ? False
~Equal     P1 P2   ? True
Less       P1 P2   ? False
Greate     P1 P2   ? True
Origin     P1      ? False
IsOnSbX    P1      ? False
IsOnSbY    P1      ? False
P1 Kuadran Ke      : 1
P2 Kuadran Ke      : 2
P3 Kuadran Ke      : 4
P4 Kuadran Ke      : 3
Next P1           : (5.00,4.00)
Next P2           : (-2.00,3.00)
PlusDelta P3 (2,2) : (4.00,0.00)
Miror P4 SbX      : (-2.00,2.00)
Geser P1 (2,3)    : (7.00,7.00)
Miror P2 SbX      : (-3.00,3.00)
Miror P3 SbY      : (4.00,-1.00)
Tulis P1          : (7.00,7.00)
Jarak P1 dr (0,0) : 9.90
Jarak P1 P2       : 10.77
P1 sbhm rot 30 CCW : (7.00,7.00)
P1 stlh rot 30 CCW : (-5.84,8.00)
P1 stlh rot -30 CCW : (7.00,7.00)
Koord 2 titik P1 dan P2 : (-0.50,7.50)
```

Berikut definisi dan spesifikasi fungsi yang harus selesai dan disimpan dalam **titik.py**

I.S : Initial State

F.S : Final State

#Fungsi Konstruktor

Function MakePoint(a,b)

''' MakePoint(a,b) membentuk sebuah titik dari a dan b dengan a sebagai absis dan b sebagai ordinat
'''

#Selektor

Function GetAbsis()→Real

'''Fungsi untuk mengambil nilai absis
mengembalikan nilai absis dari variable global
'''

Function GetOrdinat()→Real

'''Fungsi untuk mengambil nilai ordinat
mengembalikan nilai ordinat dari variable global
Fungsi Kolektor
'''

Function SetOrdinat(newx)

'''Fungsi untuk menyetel nilai Ordinat suatu titik
di variable Global'''

Function SetAbsis(newy)

'''Fungsi untuk menyetel nilai Absis suatu titik
di variable Global
Fungsi Baca Tulis'''

Function BacaPoint()

'''Fungsi untuk membaca nilai suatu titik dari keyboard
baik nilai absis atau ordinat, perlu variable local
dalam prosedur untuk di assign ke dalam variable global'''

Function Tulis(x,y) ;

'''
{ Nilai variable global x dan y ditulis ke layar dengan format
"(x,y)" }
{ I.S. <x,y> terdefinisi }
{ F.S. <x,y> tertulis di layer dengan format "(X,Y)" }
'''
'''Operasi aritmatika'''

Function AddP(x1,y1,x2,y2)→Real

'''
{ Menghasilkan titik bernilai <x1,y1> +- <x2,y2>2 }
{ Buatlah spesifikasi penambahan dua buah titik }
'''

Function MinP(x1,y1,x2,y2)→Real

'''
{ Menghasilkan titik bernilai <x1,y1> - <x2,y2>2 }
{ Buatlah spesifikasi pengurangan dua buah titik }
'''

```

Function MulDot(x1,y1,x2,y2)→Real
'''
{ Operasi perkalian <x1,y1>*<x2,y2> : melakukan operasi dot
product}
{ *** Kelompok operasi relasional terhadap titik *** }
'''
'''Posisi'''
Function IsOrigin(x,y)→Boolean
'''
IsOrigin(x,y) benar jika <x,y> adalah titik origin yaitu titik
<0,0>
'''
Function IsOnSbX()→Boolean
'''Menghasilkan true jika <x,y> terletak pada sumbu X'''
Function IsOnSbY()→Boolean
'''Menghasilkan true jika <x,y> terletak pada sumbu Y'''
Function Kuadran()→Integer
'''
{ Menghasilkan kuadran dari <x,y>: 1, 2, 3, atau 4 }
{ Prekondisi : <x,y> bukan Titik Origin, }
{ dan <x,y> tidak terletak di salah satu sumbu }
'''

'''Operasi Tipe'''
Function NextX()→Real
'''
Mengirim salinan <x,y> dengan absis ditambah satu
'''
Function NextY()→Real
'''
Mengirim salinan <x,y> dengan ordinat ditambah satu
'''
Function PlusDelta(deltax,deltay)
'''
Menghasilkan <x,y> yang absisnya adalah Absis(x) + deltaX dan
ordinatnya adalah Ordinat(y) + deltaY
'''
Function MirrorOf(sbx)→Real
'''
Menghasilkan <x,y> yg dicerminkan dg nilai SbX dan SbY
Jika SbX bernilai true, maka dicerminkan terhadap sumbu X
Jika SbY bernilai true, maka dicerminkan terhadap sumbu Y
'''
Function JarakPst()→Real
'''
Menghitung jarak <x,y> ke (0,0)
'''
Function Geser(deltax,deltay)

```

```

'''
{ I.S. x dan y sudah terdefinisi }
{ F.S. x dan y digeser sebesar deltaX & ordinatnya sebesar
deltaY }
'''
Function GeserSbX()
'''
{ I.S. x dan y sudah terdefinisi }
{ F.S. x dan y di Sumbu X dengan ordinat yang sama dgn semula. }
{ Proses : x dan y digeser ke sumbu X. }
'''
Function GeserSbY()
'''
{ I.S. x dan y sudah terdefinisi }
{ F.S. x dan y di Sumbu Y dengan ordinat yang sama dgn semula. }
{ Proses : x dan y digeser ke sumbu Y. }
'''
Function MirrorP(sbx)
'''
{ I.S. x dan y sudah terdefinisi }
{ F.S. x dan y dicerminkan tergantung nilai SbX atau SbY }
{ Jika SbX true maka dicerminkan terhadap sumbu X }
{ Jika SbY true maka dicerminkan terhadap sumbu Y }
'''
Function Putar(sudut)
'''
{ I.S. x dan y sudah terdefinisi }
{ F.S. x dan y diputar sebesar sudut derajat }
'''

```