

~\Documents\Kode_24060124120013\Kode_24060124120013\2-garis.py

```
1  """
2  Program    : Tipe Garis
3  Deskripsi  : Menentukan tipe garis yang terdiri dari dua titik yaitu titik awal dan titik akhir
4  NIM>Nama   : 24060124110142/Muchammad Yuda Tri Ananda
5  Tanggal   : 29/09/2024
6
7  ****
8  DEFINISI DAN SPESIFIKASI TYPE
9  type point : < x : integer, y : integer >
10     <x, y> adalah sebuah titik yang mempunyai absis x dan ordinat y
11  type garis : < p1 : point, p2 : point >
12     <p1, p2> adalah sebuah garis yang terdiri dari dua titik yaitu titik awal (p1) dan titik
    akhir (p2)
13
14  ****
15  DEFINISI DAN SPESIFIKASI SELEKTOR
16
17  absis : point ---> integer
18     absis(P) mengembalikan nilai x atau absis dari titik P
19  ordinat : point ---> integer
20     ordinat(P) mengembalikan nilai y atau ordinat dari titik P
21  StartP : garis ---> point
22     StartP(G) mengembalikan titik awal dari garis G
23  EndP : garis ---> point
24     EndP(G) mengembalikan titik akhir dari garis G
25
26  ****
27  DEFINISI DAN SPESIFIKASI KONSTRUKTOR
28
29  MakePoint: 2 integer ---> point
30     MakePoint(x, y) membuat tipe point dengan absis x dan ordinat y
31  MakeGaris : 2 point ---> garis
32     MakeGaris(p1,p2) membuat tipe garis dengan titik awal p1 dan titik akhir p2
33
34  ****
35  DEFINISI DAN SPESIFIKASI OPERATOR
36
37  PanjangGaris : garis ---> real
38     PanjangGaris(G) menghitung panjang garis G
39  Gradien : garis ---> real
40     Gradien(G) menghitung gradien garis G
41  jarak : 2 point ---> real
42     jarak(p1, p2) mengembalikan jarak antara p1 dengan p2
43
44  ****
45  DEFINISI DAN SPESIFIKASI FUNGSI PREDIKAT
46
47  IsSejajar : 2 garis ---> boolean
```

```

48     IsSejajar(g1,g2) menentukan apakah garis g1 dan g2 sejajar
49 IsTegakLurus : 2 garis ---> boolean
50     IsTegakLurus(g1, g2) menentukan apakah garis g1 tegak lurus dengan garis g2
51
52 *****
53 DEFINISI DAN SPESIFIKASI FUNGSI ANTARA
54
55 fx2 : integer ---> integer
56     fx2(x) memberikan nilai kuadrat dari x
57
58 *****
59
60 REALISASI
61 *****
62 """
63
64 from math import sqrt
65
66
67 def jarak(P1, P2):
68     return sqrt(fx2(absis(P2) - absis(P1)) + fx2(ordinat(P2) - ordinat(P1)))
69
70
71 def fx2(x):
72     return x * x
73
74
75 def MakePoint(x, y):
76     return [x, y]
77
78
79 def absis(P):
80     return P[0]
81
82
83 def ordinat(P):
84     return P[1]
85
86
87 # Fungsi Utama
88
89
90 # Konstruktor
91 def MakeGaris(p1, p2):
92     return [p1, p2]
93
94
95 # Selektor
96 def StartP(g):
97     return g[0]

```

```

98
99
100 def EndP(g):
101     return g[1]
102
103
104 # Operator
105 def PanjangGaris(g):
106     return jarak(StartP(g), EndP(g))
107
108
109 def Gradien(g):
110     return (ordinat(EndP(g)) - ordinat(StartP(g))) / (absis(EndP(g)) - absis(StartP(g)))
111
112
113 # Predikat
114 def IsSejajar(g1, g2):
115     return Gradien(g1) == Gradien(g2)
116
117
118 def IsTegakLurus(g1, g2):
119     return Gradien(g1) * Gradien(g2) == -1
120
121
122 """
123 APLIKASI
124 *****
125 """
126
127 print(PanjangGaris(MakeGaris(MakePoint(0, 0), MakePoint(3, 4))))
128 print(Gradien(MakeGaris(MakePoint(0, 0), MakePoint(3, 4))))
129 print(
130     IsSejajar(
131         MakeGaris(MakePoint(0, 0), MakePoint(3, 4)),
132         MakeGaris(MakePoint(-3, -4), MakePoint(0, 0)),
133     )
134 )
135 print(
136     IsTegakLurus(
137         MakeGaris(MakePoint(-3, 1), MakePoint(3, 0)),
138         MakeGaris(MakePoint(1, 3), MakePoint(0, -3)),
139     )
140 )
141

```