

## Tipe Bentuk Pecahan Campuran

```
1 # Nama File : pecahan_campuran.py
2 # Deskripsi : membuat tipe bentuk pecahan campuran beserta konstruktor dan selektornya
3 # Pembuat : Muhammad Fikri-24060124130069
4 # Tanggal : 11 Oktober 2024
5
6
7 def bil (P) :
8     return P[0]
9
10 def num (P) :
11     return P[1]
12
13 def denum (P) :
14     return P[2]
15
16 def MakePecahanc (b, n, d) :
17     return [b, n, d]
18
19 def MakePecahan (n, d) :
20     return [n, d]
21
22 def KonversiPecahan (P) :
23     return MakePecahan(
24         bil(P) * denum(P) + num(P), denum(P)
25     )
26
27 def KonversiReal (P) :
28     return numc(P) / denum(P)
29
30 def AddP (P1,P2) :
31     return MakePecahanc(
32         (numc(P1) * denum(P2) + numc(P2) * denum(P1)) // (denum(P1) * denum(P2)),
33         (numc(P1) * denum(P2) + numc(P2) * denum(P1)) - ((denum(P1) * denum(P2)) *
34         (numc(P1) * denum(P2) + numc(P2) * denum(P1)) // (denum(P1) * denum(P2))),
35         denum(P1) * denum(P2)
36     )
37
38 def SubP (P1, P2) :
39     return MakePecahanc(
40         (numc(P1) * denum(P2) - numc(P2) * denum(P1)) // (denum(P1) * denum(P2)),
41         (numc(P1) * denum(P2) - numc(P2) * denum(P1)) - ((denum(P1) * denum(P2)) *
42         (numc(P1) * denum(P2) - numc(P2) * denum(P1)) // (denum(P1) * denum(P2))),
43         denum(P1) * denum(P2)
44     )
45
46 def DivP (P1,P2) :
47     return MakePecahanc(
48         (numc(P1) * denum(P2)) // (denum(P1) * numc(P2)),
49         (numc(P1) * denum(P2)) - (denum(P1) * numc(P2)) * ((numc(P1) * denum(P2)) // (denum(P1) * numc(P2))),
50         (denum(P1) * numc(P2))
51     )
52
53 def MulP (P1,P2) :
54     return MakePecahanc(
55         (numc(P1) * numc(P2)) // (denum(P1) * denum(P2)),
56         (numc(P1) * numc(P2)) - (denum(P1) * denum(P2)) * ((numc(P1) * numc(P2)) // (denum(P1) * denum(P2))),
57         denum(P1) * denum(P2)
58     )
```

```
60 def numc (P) :
61     return bil(P) * denum(P) + num(P)
62
63 def IsEqP (P1,P2) :
64     return numc(P1) * denum(P2) == numc(P2) * denum(P1)
65
66 def IsLtP (P1,P2) :
67     return numc(P1) * denum(P2) < numc(P2) * denum(P1)
68
69 def IsGtP (P1,P2) :
70     return numc(P1) * denum(P2) > numc(P2) * denum(P1)
71
72
73 # APLIKASI
74
75 P1 = MakePecahanc(2, 1, 2)
76 P2 = MakePecahanc(1, 1, 2)
77 print(KonversiPecahan(P1)) # [5, 2]
78 print(KonversiPecahan(P2)) # [3, 2]
79 print(KonversiReal(P1))    # 2.5
80 print(KonversiReal(P2))    # 1.5
81 print(AddP(P1, P2))        # [4, 0, 4]
82 print(SubP(P1, P2))        # [1, 0, 4]
83 print(DivP(P1, P2))        # [1, 4, 6]
84 print(MulP(P1, P2))        # [3, 3, 4]
85 print(IsEqP(P1, P2))       # False
86 print(IsLtP(P1, P2))       # False
87 print(IsGtP(P1, P2))       # True
```

## Tipe Bentuk Garis

```
1  # Nama File : garis.py
2  # Deskripsi : membuat tipe bentuk garis beserta konstruksi dan selektornya
3  # Pembuat   : Muhammad Fikri-24060124130069
4  # Tanggal   : 11 Oktober 2024
5
6
7  from math import sqrt
8
9  def absis (P) :
10     return P[0]
11
12  def ordinat (P) :
13     return P[1]
14
15  def garisAwal (P) :
16     return P[0]
17
18  def garisAkhir (P) :
19     return P[1]
20
21  def makePoint (a, b):
22     return [a, b]
23
24  def makeGaris (P1, P2) :
25     return [P1, P2]
26
27  def Gradien (G) :
28     return (ordinat(garisAkhir(G)) - ordinat(garisAwal(G))) / (absis(garisAkhir(G)) - absis(garisAwal(G)))
29
30  def PanjangGaris (G):
31     return sqrt(
32         fx2(absis(garisAkhir(G)) - absis(garisAwal(G))) +
33         fx2(ordinat(garisAkhir(G)) - ordinat(garisAwal(G)))
34     )
35
36  def fx2 (x):
37     return x*x
38
39  def IsSejajar (G1, G2) :
40     return Gradien(G1) == Gradien(G2)
41
42  def IsTegakLurus (G1, G2) :
43     return Gradien(G1) * Gradien(G2) == -1
44
45
46  # APLIKASI
47
48  p1 = makePoint(4.0, 2.0)
49  p2 = makePoint(5.0, 4.0)
50  p3 = makePoint(-4.0, 2.0)
51  p4 = makePoint(-8.0, 4.0)
52
53  g1 = makeGaris(p1, p2)
54  g2 = makeGaris(p3, p4)
55
56  print(Gradien(g1)) # 2.0
57  print(Gradien(g2)) # -0.5
58  print(IsSejajar(g1, g2)) # False
59  print(IsTegakLurus(g1, g2)) # True
```