

## 9 関数を定義する

Pascal に用意されていない関数を自分で定義して使うことを学びます。

### 9.1 SankakuhiNoHyou2.dpr (三角比の表, 改訂版)

三角関数は, Pascal には正弦 Sin と余弦 Cos が用意されていますが, 正接 Tan はありません。また, Sin, Cos の角の単位は度 (°) ではなく弧度 (ラジアン) です。

正接 Tan と, 角の単位が度の正弦 SinDo, 余弦 CosDo, 正接 TanDo を自分で定義して三角比の表を作るプログラムです。

```

program SankakuhiNoHyou2; // 学生証番号 氏名
{$APPTYPE CONSOLE}
uses
  SysUtils;

function Tan(T : Real) : Real;
  { tan }
begin
  Tan := Sin(T)/Cos(T);
end; {Tan}

function SinDo(A : Real) : Real;
  { sin ° }
begin
  SinDo := Sin(A/180*Pi);
end; {SinDo}

function CosDo(A : Real) : Real;
  { cos ° }
begin
  CosDo := Cos(A/180*Pi);
end; {CosDo}

function TanDo(A : Real) : Real;
  { tan ° }
begin
  TanDo := Tan(A/180*Pi);
end; {TanDo}

var
  Alpha1 : Integer; // 0 ~ 44
  Alpha2 : Integer; // 45 ~ 89

begin
  Write(' ':7, '':2, 'sin ':8, 'cos ':8, 'tan ':8);
  Write(' ':7, '':2, 'sin ':8, 'cos ':8, 'tan ':8);
  WriteLn;
  WriteLn;
  Alpha1 := 0;
  repeat
    Alpha2 := Alpha1+45;
    Write(Alpha1:7, '° ', SinDo(Alpha1):8:4, CosDo(Alpha1):8:4, TanDo(Alpha1):8:4);
    Write(Alpha2:7, '° ', SinDo(Alpha2):8:4, CosDo(Alpha2):8:4, TanDo(Alpha2):8:4);
  
```

```

WriteLn;
if Alpha1 mod 5 = 4
  then WriteLn;
  Alpha1 := Alpha1+1;
until Alpha1 > 44;
Write(' ':7, '':2, 'sin ':8, 'cos ':8, 'tan ':8);
Write(' ':7, '':2, 'sin ':8, 'cos ':8, 'tan ':8);
WriteLn;
ReadLn;
end.

```

## 9.2 Sankakukei1.dpr (二辺挟角から三角形を決定する)

### 9.2.1 実行例

二辺挟角から三角形を決定する

```

辺 CA [cm] ? 17.32
辺 AB [cm] ? 14.14
角 A [°] ? 75

```

```

角 A = 75.00 [°]
角 B = 60.00 [°]
角 C = 45.00 [°]
辺 BC = 19.32 [cm]
辺 CA = 17.32 [cm]
辺 AB = 14.14 [cm]

```

Push Enter Key

### 9.2.2 解説

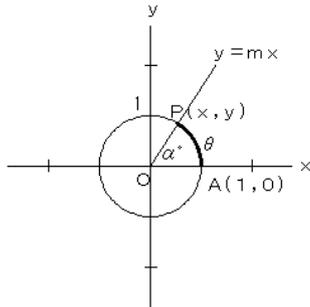
三角形 ABC の二辺と挟角 ( $b = CA$ ,  $c = AB$ ,  $A = \angle A$ ) が与えられると, 三角形が定まります。すなわち角  $A, B, C$  と辺  $a, b, c$  が定まります。

公式 9.1 (余弦定理)

$$\begin{aligned}
 a^2 &= b^2 + c^2 - 2bc \cos A \\
 \therefore a &= \sqrt{b^2 + c^2 - 2bc \cos A} \\
 \cos B &= \frac{c^2 + a^2 - b^2}{2ca} \\
 \therefore B &= \cos^{-1} \left( \frac{c^2 + a^2 - b^2}{2ca} \right) \\
 C &= 180^\circ - A - B
 \end{aligned}$$

二辺挟角を入力して三角三辺を出力するプログラムを作りたいのですが, それには逆余弦関数  $\cos^{-1}$ (アークコサイン) が必要です。しかし, Pascal に用意されている逆三角関数は逆正接関数  $\tan^{-1}$  を計算する ArcTan だけです。そのため, 逆三角関数の定義を参考にして, ArcCos を作ることにします。

定義 9.1 逆三角関数



$$\theta = \cos^{-1}(x) \Leftrightarrow \cos \theta = x \quad \text{かつ} \quad 0 \leq \theta \leq \pi$$

$$\theta = \sin^{-1}(y) \Leftrightarrow \sin \theta = y \quad \text{かつ} \quad -\frac{\pi}{2} \leq \theta \leq \frac{\pi}{2}$$

$$\theta = \tan^{-1}(m) \Leftrightarrow \tan \theta = m \quad \text{かつ} \quad -\frac{\pi}{2} < \theta < \frac{\pi}{2}$$

特に  $\theta$  の範囲に注意しないとイケません。

$$m = \tan \theta = \frac{\sin \theta}{\cos \theta} = \frac{y}{x} = \frac{\sqrt{1-x^2}}{x}$$

$$\therefore \cos^{-1}(x) = \begin{cases} \tan^{-1}(m) & (x > 0 \text{ のとき}) \\ \frac{\pi}{2} & (x = 0 \text{ のとき}) \\ \tan^{-1}(m) + \pi & (x < 0 \text{ のとき}) \end{cases}$$

function ArcCos(X : Real) : Real; の実行部は, 9.1 節の例のように 1 行では書けません。if を使って場合分けして T ( $\theta$  を表す変数) の値を計算し, 最後に関数の値として ArcCos に代入します。

ArcCos が返す値は単位がラジアンなので, さらに単位が度の角を返す ArcCosDo も定義します。

9.2.3 プログラム

```

program Sankakukei1; // 学生証番号 氏名
{$AppType Console}
uses
  SysUtils;

function ArcCos(X : Real) : Real;
  { ArcCos := (cos = X & 0          となる角  }
  { 単位はラジアン                  }
  { -1 <= X <= 1 でないとエラーになる }
var
  Y : Real; // sin
  M : Real; // tan
  T : Real; //
begin
  if X = 0
  then begin
    T := Pi/2;
  end
  else begin
    Y := Sqrt(1-Sqr(X));
    M := Y/X;
    T := ArcTan(M); // - /2 < < /2
    if X < 0
  end;
end;
    
```

```

        then T := T+Pi;
      end;
      ArcCos := T;
    end; {ArcCos}

function ArcCosDo(X : Real) : Real;
  { ArcCosDo := (cos ° = X & 0      180 となる角  }
  { -1 ≤ X ≤ 1 でないとエラーになる }
begin
  ArcCosDo := ArcCos(X)/Pi*180;
end; {ArcCosDo}

function CosDo(A : Real) : Real;
  { cos ° }
begin
  CosDo := Cos(A/180*Pi);
end; {CosDo}

var
  KakuA,KakuB,KakuC : Real; // 角
  HenBC,HenCA,HenAB : Real; // 辺
  BC2Jou           : Real; // BC^2 の値
  CosB             : Real; // cos B の値

begin {Main}
  WriteLn('二辺挟角から三角形を決定する');
  WriteLn;
  Write('辺 CA [cm] ? ');
  ReadLn(HenCA);
  Write('辺 AB [cm] ? ');
  ReadLn(HenAB);
  Write('角 A [°] ? ');
  ReadLn(KakuA);
  WriteLn;
  BC2Jou := Sqr(HenCA)+Sqr(HenAB)-2*HenCA*HenAB*CosDo(KakuA);
  HenBC := Sqrt(BC2Jou);
  CosB := (Sqr(HenAB)+Sqr(HenBC)-Sqr(HenCA))/(2*HenAB*HenBC);
  KakuB := ArcCosDo(CosB);
  KakuC := 180-KakuA-KakuB;
  WriteLn('角 A = ', KakuA:6:2, ' [°]');
  WriteLn('角 B = ', KakuB:6:2, ' [°]');
  WriteLn('角 C = ', KakuC:6:2, ' [°]');
  WriteLn('辺 BC = ', HenBC:6:2, ' [cm]');
  WriteLn('辺 CA = ', HenCA:6:2, ' [cm]');
  WriteLn('辺 AB = ', HenAB:6:2, ' [cm]');
  WriteLn;
  Write('Push Enter Key ');
  ReadLn;
end.

```

### 9.3 KyokuKeishiki.dpr (極形式)

複素数  $z = x + yi$  を  $r(\cos \theta + i \sin \theta)$  の形で表したものを極形式と言います。  $z$  の絶対値  $r$  と偏角  $\theta$  は次のように  $x, y$  から求められます。

$$r = \sqrt{x^2 + y^2}$$

$$\theta = \begin{cases} \frac{\pi}{2} & (x = 0 \text{ かつ } y \geq 0 \text{ のとき}) \\ -\frac{\pi}{2} & (x = 0 \text{ かつ } y < 0 \text{ のとき}) \\ \tan^{-1}\left(\frac{y}{x}\right) & (x > 0 \text{ のとき}) \\ \tan^{-1}\left(\frac{y}{x}\right) + \pi & (x < 0 \text{ かつ } y \geq 0 \text{ のとき}) \\ \tan^{-1}\left(\frac{y}{x}\right) - \pi & (x < 0 \text{ かつ } y < 0 \text{ のとき}) \end{cases}$$

#### 例 9.1

直交形式	極形式
$5 + 5i$	$= 5\sqrt{2}(\cos 45^\circ + i \sin 45^\circ)$
$-4\sqrt{3} + 4i$	$= 8(\cos 150^\circ + i \sin 150^\circ)$
$-2 - 2\sqrt{3}i$	$= 4(\cos(-120^\circ) + i \sin(-120^\circ))$
$-3i$	$= 3(\cos(-90^\circ) + i \sin(-90^\circ))$
$3 - 4i$	$= 5(\cos(-53.1^\circ) + i \sin(-53.1^\circ))$

#### 9.3.1 実行例

複素数  $x + y i$  を極形式に変換します

x と y [0 0 を入れたら終わります] ? 5 5  
 $5.00 + 5.00 i = 7.07(\cos 45.0^\circ + i \sin 45.0^\circ)$

x と y [0 0 を入れたら終わります] ? -6.93 4  
 $-6.93 + 4.00 i = 8.00(\cos 150.0^\circ + i \sin 150.0^\circ)$

x と y [0 0 を入れたら終わります] ? -2 -3.46  
 $-2.00 + -3.46 i = 4.00(\cos -120.0^\circ + i \sin -120.0^\circ)$

x と y [0 0 を入れたら終わります] ? 0 -3  
 $0.00 + -3.00 i = 3.00(\cos -90.0^\circ + i \sin -90.0^\circ)$

x と y [0 0 を入れたら終わります] ? 3 -4  
 $3.00 + -4.00 i = 5.00(\cos -53.1^\circ + i \sin -53.1^\circ)$

x と y [0 0 を入れたら終わります] ? 0 0

#### 9.3.2 問題

このプログラムを作成しなさい。ただし、絶対値を求める関数 `Zettaichi(X,Y : Real) : Real` と偏角 (単位は度) を求める関数 `HenkakuDo(X,Y : Real) : Real` を定義して用いること。

```
program KyokuKeishiki; // 学生証番号 氏名
{$APPTYPE CONSOLE}
uses SysUtils;

function HenkakuDo(X,Y : Real) : Real;
    {  $x + yi$  の偏角 , 単位は度 }
    var

begin

end; {HenkakuDo}

function Zettaichi(X,Y : Real) : Real;
    {  $x + yi$  の絶対値  $r$  }
    begin

end; {Zettaichi}

var

begin

end.
```