

この時間のねらい: ① 移動・伸縮・回転させるための関数を使えるようになる。③

今回の内容は、回転の `rotate()` 関数になります。この関数を使うためには本来は数学Ⅱで学ぶ「弧度法」の知識が必要になります。そのため、可哀想ですが、授業はプログラミングなのに数学をやらなければいけません。ということで、以下から始めていきます。

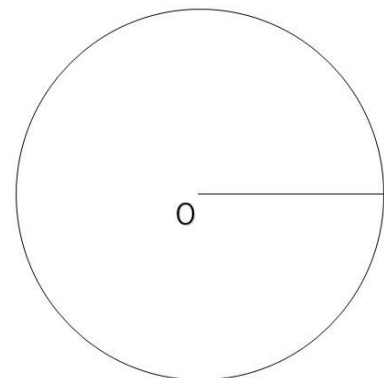
#### ・弧度法とは

皆さんが親しんでいる角度の表記法は  $60^\circ$  という表記方法だと思います。これは **度数法** と言い、**角度の程度** を数値で表す方法になります。度数法のほかに角度を表記するための手法として、タイトルにもある **弧度法** というものがあります。弧度法での単位は **ラジアン** もしくは、**弧度** の 2 種類があります。ただ、弧度という単位を使う人は少ないイメージがあります。なので、覚えておくのは **ラジアン** でよいでしょう。そして、基本的に弧度法の値に活用するものは  **$\pi$  (円周率)** になります。それでは、どのように定義をしていくのかを見てみましょう。

#### ・弧度法の定義方法

右の円を考えます。半径は 1 です。ここで適当な 2 点 A, B を書きましょう。その時の  **$\widehat{AB}$**  の長さを  **$\theta$**  とするとその時の角度は  **$\theta$  ラジアン** となります。要するに **弧の長さ** によって角度を定義するという考え方になります。

ここで、基準を決めるため、この円の半円の弧の長さを求めましょう。弧の長さの求め方は覚えていますか？中学校の頃に学んだ内容だと思いますが、忘れている人が多いかもしれません。以下の公式を使います。



弧の長さ  $l$  を求める公式:  $l = 2\pi r \frac{\theta}{360^\circ}$  ( $r$ : 半径,  $\theta$ : 中心角)

半円の場合、 **$\theta = 180^\circ$**  であるので、最終的な結果は  **$l = \pi$**  になります。このことから、基準として  **$180^\circ = \pi$  ラジアン** という等式が成り立ちます。これをもとにして、例えば、 **$90^\circ = \frac{\pi}{2}$  ラジアン**,  **$60^\circ = \frac{\pi}{3}$  ラジアン**,  **$30^\circ = \frac{\pi}{6}$  ラジアン** のように計算することができる。以下の練習問題によって度数法から弧度法への変換を練習しましょう。

#### 練習問題

- (1)  $45^\circ = \frac{\pi}{4}$  ラジアン    (2)  $120^\circ = \frac{2}{3}\pi$  ラジアン    (3)  $135^\circ = \frac{3}{4}\pi$  ラジアン  
 (4)  $1^\circ = \frac{\pi}{180}$  ラジアン    (5)  $160^\circ = \frac{8}{9}\pi$  ラジアン    (6)  $360^\circ = 2\pi$  ラジアン

表の内容にて、弧度法への変換がなんとかできるようになったかと思います。では、続いて rotate()関数の使い方になります。実際にプログラムに打ち込む方法としては、9 回目のプリントにあるため、必要があれば参照してください。

ただ、この関数でやっていることを知らないと、奇妙なバグが起きる可能性があります。それは、回転させたい図形を画面外へ移動させてしまうというものです。今回学んでいる、rotate()関数は、**座標**自体を回転させる関数になっています。そのため、**人間側**が座標を意識しておかないと考えづらい内容になっています。以上の内容を確認・修正できるように、次からの演習問題をやってみましょう。

#### 演習問題 1

```
sketch_210526a
1 void setup() {
2   //ウィンドウを表示させる関数を記述
3 }
4
5 void draw() {
6   //三角形を描画する関数を記述(大きさは指定しない)
7 }
```

#### 演習問題 2

演習問題 1 で作成したプログラムに本日学んだ rotate()関数を利用して、90° だけ反時計回りに回転させる命令を記述しなさい。

#### 演習問題 3

演習問題 2 のプログラムを実行すると、三角形が画面上にない状態になっている。これを改善させるため、translate()関数を利用して、三角形内部の位置を原点に設定することで、画面内に表示させられるように改良を行いなさい。

(場合によって、変数宣言をすることでやりやすい状態になります。)