

論理思考とプログラミング

第9回

contents

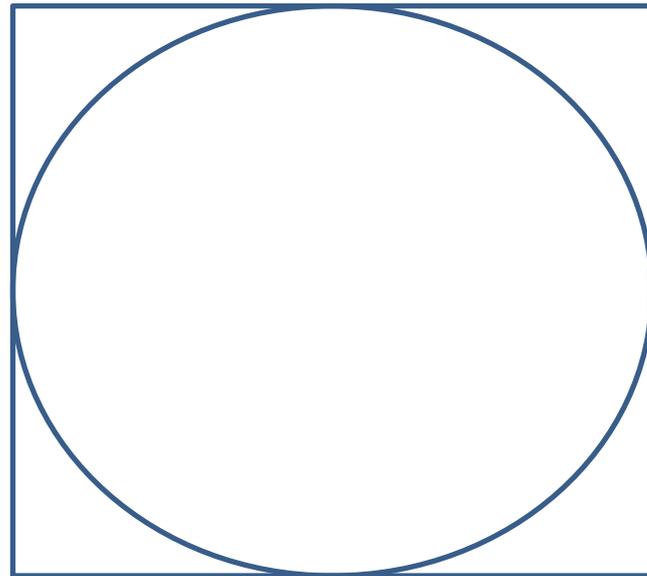
- モンテカルロ法の概要と実装
- 二次元配列
- Webサービスの概要
 - google map を用いたプログラムの作成

モンテカルロ法

教材7-3-5

モンテカルロ法を用いて円周率を求める

- 乱数を使ったシミュレーションを繰り返すことで近似解を得る手法
- 一辺の長さ2の正方形と、それに内接する半径1の円を考える
- ランダムにN個の点を正方形内に打つ
- 円の内部に打たれた点の数をkとする
- 半径1の円の面積は π である(πr^2)
- 一辺の長さが2の正方形の面積は4である
- Nが十分に大きい時、Nとkの比は正方形の面積と円の面積の比に近似する
- つまり、 $N:k \doteq 4:\pi$
- ゆえに、 π はほぼ、 $4k/N$



配布したプログラムの解説

- 作成する関数は2つ

- calcPi()

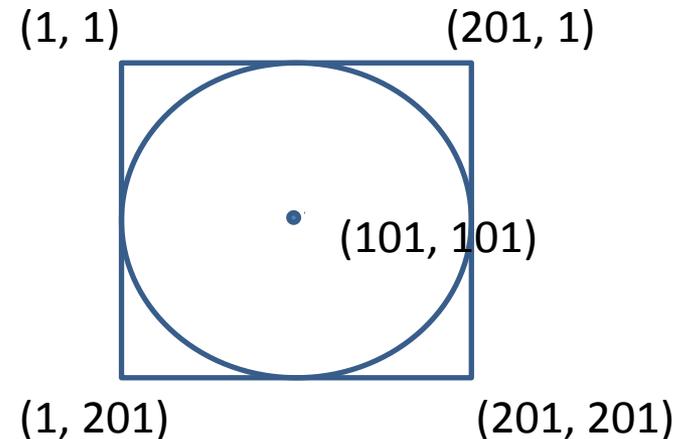
- 正方形内部にランダムに点を打つ
 - 打った点が正方形に内接する円の内部にあるか判定する
 - 打った数と円の内部に入った数から π を計算する

- calcPiReset()

- 正方形とそれに内接する円を再描画する
 - 点を打った回数を0に戻す
 - 始めに一度実行されることに注意

calcPiReset()

- 半径100ドットの円を描画する
 - `arc(x, y, r, startAngle, endAngle, anticlockwise)`を使って円弧を描画する
 - `x, y`は中心点の座標、`r`は半径
 - 今回は、`x = 101, y = 101, r = 100`
 - `startAngle, endAngle`は円弧の中心角(ラジアンで指定)
 - 今回は円を描画するので中心角は360度
 - ラジアンは度数 $\div 180 \times \pi$ なので、`startAngle`は0、`endAngle`は $360/180 \times \pi$ 、即ち、`Math.Pi*2`
 - `anticlockwise`は反時計回りに描画するか否かを`true, false`で指定する。`true`が反時計回り
- 半径200ドットの正方形を描画する
 - `lineTo()`を使って描画する

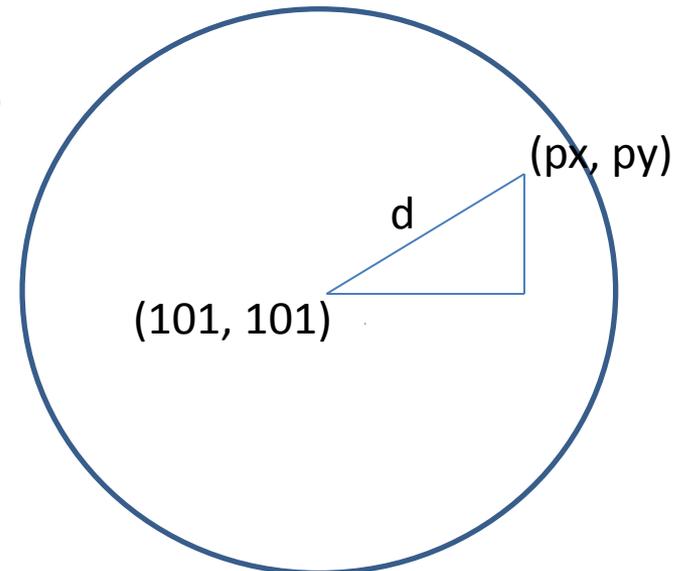


calcPi()

- 正方形内にランダムな点を打つ
 - 1度クリックする度に1000個の点を打つとする(for文)
 - 点を打った回数をnaとし、for文中でna = na + 1を実行する
 - 点の座標をpx, pyとし、以下のようにランダムな座標を得る(pxの場合)
 - $px = \text{Math.random()} * 2 * r + (x - r);$
 - random()は0～1までの実数をランダムに生成する
 - 0の場合、 $0 * 2 * 100 + (101 - 100) \rightarrow 1$
 - 0.5の場合、 $0.5 * 2 * 100 + (101 - 100) \rightarrow 101$
 - 1の場合、 $1 * 2 * 100 + (101 - 100) \rightarrow 201$
 - fillRect()を使って、px, pyを基点に一辺の長さ1ドットの正方形を描画する(点のつもり)

calcPi()

- 打った点が円の内部にあるかを判定する
 - 円の中心から、打たれた点までの距離を d とすると、ピタゴラスの定理により、 $d^2 = (px - x)^2 + (py - y)^2$
 - d^2 を pz とすると、 $pz = (px - x)^2 + (py - y)^2$
 - 打った点が円の内部にあるということは、 d が円の半径以下($d \leq r$)ということ
 - つまり、 d^2 が半径の二乗以下かどうかを調べればよい
 - よって、条件式は $pz < r^2$ となる
- 点が円の内部に入った数(nc)をカウントする
 - $pz < r^2$ ならば、 nc を1増やす



calcPi()

- π の値を計算する

- 打った点の総数:円内部の点の数: \div 一辺の長さ200の正方形の面積:半径100の円の面積
- つまり、 $na:nc = 200*200:100*100*\pi$
- $\rightarrow (100*100)*na*\pi = (200*200)*nc$
- 即ち、 $\pi = (200*200)*nc/(100*100)*na = nc/4*na$
- 教材のプログラムでは以下のように式が書かれているが同じ事(結局は $nc/4na$)
$$nc/na * (200*200)/(100*100)$$

二次元配列

二次元配列(1)

- 配列とは・・・(復習)
 - 一つの変数名に複数の要素を記憶させることができる
 - `var array = ["sato", "suzuki", "yamada", "ito"];`
 - 要素数4の配列を作ったことになる
 - `document.write(array[2]);`のように添え字番号で配列に記憶された要素を参照できる(この場合、yamadaが出力される)
 - 添え字は0から始まるので注意
 - 配列の要素数は、`array.length`で取得できる

二次元配列(2)

- 配列で名前、学籍番号、ログイン名を管理することを考える
 - name = [“akiyama”, “arai”, “kobayashi”, “uchida”];
 - student_id = [70000001, 70000002, 70000003, 70000004];
 - login_name = [“maoon”, “t10xxxya”, “t07xxxmk”, “t07xxxsu”];
- 色々めんどろ(変数名が多くなる、要素の対応付けが大変、等々)

二次元配列(3)

- 二次元配列(配列の配列)を使うと便利。以下のように配列をつくります

```
student_info = [  
  ["akiyama", 70000001, "maooon"],  
  ["arai", 70000002, "t10xxxya"],  
  ["kobayashi", 70000003, "t07xxxmk"],  
  ["uchida", 70000004, "t07xxxsu"]  
];
```

```
document.write(student_info[1][2]);
```

→2行目の3列目の要素(添え字が0から始まることに注意)が表示される

```
array[行][列]
```

二次元配列(4)

- 二次元配列の長さを取得するには・・・？
 - 単なる配列array[]にいくつかの要素が入っているかは、array.lengthで取得できたが、多次元配列の場合は以下のようにする
 - array[][]の行数(一つ目の[]に入ってる要素の数)を取得するには、array.length
 - array[i][]のi行目の列数(二つめの[])を取得するには、array[i].length

二次元配列の要素を表に書き出す(1)

- 年月日、始値、高値、安値、終値を二次元配列に格納して、以下のような表として出力したいとする

日付	始値	高値	安値	終値
2011年1月10日	82.69	83.54	82.3799	82.94
2011年1月17日	82.8	83.4899	81.8199	82.58
2011年1月24日	82.51	83.1999	81.9599	82.0999
2011年1月31日	81.98	82.4599	81.0899	82.1699
2011年2月7日	82.1999	83.37	81.75	83.26
...

二次元配列の要素を表に書き出す(2)

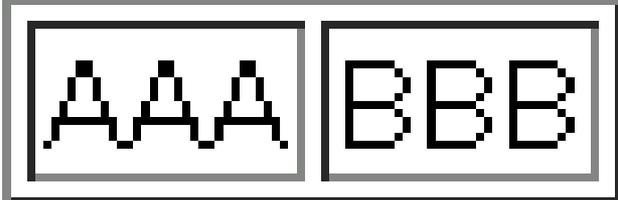
まず、配列を作る

```
var usdjpy = [  
    ["2011年1月10日",82.69,83.54,82.3799,82.94],  
    ["2011年1月17日",82.8,83.4899,81.8199,82.58],  
    ["2011年1月24日",82.51,83.1999,81.9599,82.0999],  
    ["2011年1月31日",81.98,82.4599,81.0899,82.1699],  
    ["2011年2月7日",82.1999,83.37,81.75,83.26]  
];
```

二次元配列の要素を表に書き出す(3)

- <table>タグの使い方(1行2列の表の場合)↓

```
<table border=1>  
  <tr>  
    <td>AAA</td>  
    <td>BBB</td>  
  </tr>  
</table>
```



AAA	BBB
-----	-----

二次元配列の要素を表に書き出す(4)

- <table>タグはdocument.write()で書き出す。まず、1行目の見出しを作るには以下のようにする

```
document.write("<table border=1>");
```

```
document.write("<tr><th>日付</th><th>始値</th><th>高値</th><th>安値</th><th>終値</th></tr>");
```

日付	始値	高値	安値	終値
----	----	----	----	----

二次元配列の要素を表に書き出す(5)

- ここで先ほどの配列usdjpyの見出し行と一行目だけを書き出すことにする(複数行を表示するのが課題です)

```
document.write("<table border=1><tr><th>日付</th><th>始値</th><th>高値</th><th>安値</th><th>終値</th></tr><tr>"); //見出しの行+2行目開始の<tr>タグを書き出す
var i = 0;
for(var j=0;j<=usdjpy[i].length;j=j+1){ //5回繰り返す
    document.write("<td>" + usdjpy[i][j] + "</td>"); //usdjpy[0][0]から[0][4]までの5列を表示
}
document.write("</tr></table>"); //<table>タグを閉じるのを忘れずに
```

日付	始値	高値	安値	終値
2011年1月10日	82.69	83.54	82.3799	82.94

演習問題

- 9-1-1
 - 配列は予め用意されている
 - 以下を最初に追加することで配列usdjpyをプログラム中で使うことが可能となる

```
<script src="http://ipl.sfc.keio.ac.jp/text/pro-2011-9/lib/usdjpy.js"></script>
```

- 複数行書き出すためには、繰り返しの繰り返しを書かなければならない。前スライドの1行5列書き出す処理を行数分繰り返す(43行×5列なので43回繰り返すが、行数はusdjpy.lengthで取得可能)

初めてのWebサービス

google mapsを使う

- 他の人が作成したプログラムをブラウザに読み込むことで利用することができる
- 今回はgoogle社及びyahoo社が提供しているプログラムを利用する
- 使うプログラムのURLを自分の作成した<script>タグとは別の<script>タグ内で指定する
- 教材を良く読んで下さい

演習問題

- 9-2-1
 - フォームから入力された緯度と経度をそれぞれ変数に代入する
 - これまでと同様、document.getElementById()を使う
- 9-3-1
 - yahoo アプリケーションIDを取得しても構わない人はやって下さい
 - 教材のヒントにある「HTMLを書き出す」とは・・・

```
var url =  
  "http://map.olp.yahooapis.jp/OpenLocalPlatform/V1/static?appid=mJ7ZXoSxg66JSukJMq9bRn  
  hc2KpU8qYwrxOnU2sP0uK6BNoVL5KNXHuV.0b4RF4ZjuRvcQ--&lat=" + ido.value + "&lon=" +  
  keido.value + "&z=" + zoom + "&width=800&height=800";  
document.write("");
```
- 9-3-2
 - フォームから拡大率を入力し、変数に代入する(9-2-1と同じ)
 - 拡大率が1以下の場合と20以上の場合についてif文を用いた場合分けを行う
- 9-4-2
 - bが0だった場合をif文で場合分けする
 - 商の欄をクリアするには、divElement.value = "";

今日の演習問題

- 9-1-1
- 9-2-1
- (9-3-1) yahoo IDを作成した人のみ
- 9-3-2
- 9-4-2
- 9-4-3