

ビュフォンの針の試行を行うプログラムである。

ビュフォンの針とは、

もし床に平行な線をひいて、そこに針を落すならば、2本の線と針が交差する確率はどのようになるか(<http://ja.wikipedia.org/wiki/ビュフォンの針>)という問題である、

針の長さを1, 平行な線の幅をtとした時、

これらが交差する確率は、 $2t/\pi$ となる。

そのため、試行の結果から円周率の近似値をもとめることができることが知られている。

この試行を1259712回行った際の実行結果

```
total: 1259712.0
cross: 600563.0
approximate PI: 3.1463276958453985
```

---主要な処理---

```
Random rnd = new Random();
double needlelength=3;
double borderwidth=4;
double degree = 0;
double fallpoint = 0;
double total = 0;
double cross = 0;;

//針が、縞のどの地点に落ちるか。
//"0"から"border_width"までのどこかに落ちる。
//針を落とした回数のうち針が縞を超えた回数の比を見る。
for(int i=0;i<108*108*108;i++){
    degree = Math.toRadians(360*rnd.nextDouble());
    fallpoint = rnd.nextDouble()*borderwidth;

    if(needlelength*Math.sin(2*degree)+fallpoint>borderwidth||
needlelength*Math.sin(2*degree)+fallpoint<0){
        cross++;
    }
    total++;
}
System.out.println("total: "+total);
System.out.println("cross: "+cross);
double rate = cross/total;
//System.out.println(rate);
double answer = (2*needlelength)/(rate*borderwidth);
System.out.println("approximate PI: "+answer);
```