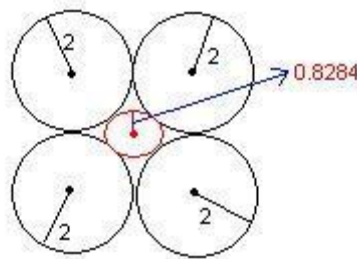


BÀI TẬP THÊM

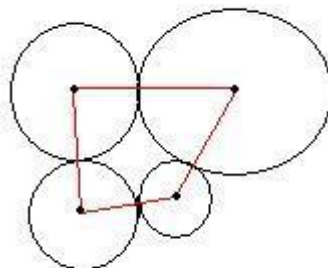
Bạn Nam có 1 bộ sưu tập tiền xu gồm N đồng xu ($N > 2$) có hình tròn, đồng xu thứ i có bán kính $r[i]$ (là số thực và dương). Bạn Nam muốn xếp chúng thành 1 vòng tròn, 2 đồng xu kề nhau thì tiếp xúc nhau. Đồng thời phải tồn tại 1 đồng xu thứ $N+1$ sao cho nếu đặt nó vào giữa vòng tròn thì nó tiếp xúc hết với N đồng xu kia. Bài toán đặt ra là hãy tính bán kính của đồng xu thứ $N+1$ này!

Ví dụ: với $N = 4$, $r[1] = r[2] = r[3] = r[4] = 2$, thì kết quả là 0.8284 như hình dưới đây:



Quả thực lần đầu tiên đọc cái đề mình hoàn toàn mất phương hướng, không biết phải làm sao cho đến khi được gợi ý là dùng **Tìm kiếm nhị phân**.

Giả sử kích thước của đồng xu cần tìm là r_{KQ} , trước hết ta cần tính xem nó nằm trong khoảng nào? Hay nói cách khác là tìm ngưỡng dưới và ngưỡng trên của nó. Ngưỡng dưới dễ thấy bằng 0, ngưỡng trên tính như sau:

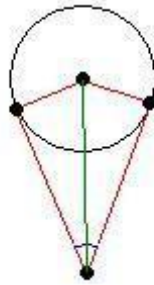


Xem hình vẽ trên, rõ ràng chu vi của đồng xu cần tìm phải nhỏ hơn chu vi của đa giác lồi (màu đỏ) nối N tâm đồng xu đã cho. Chu vi đa giác dễ thấy $= 2 * (r[1] + r[2] + \dots + r[N])$, từ công thức tính chu vi đường tròn ta suy ra ngưỡng trên là:

$$2 * (r[1] + r[2] + \dots + r[N]) / (2 * \pi) = (r[1] + r[2] + \dots + r[N]) / \pi$$

Đã có ngưỡng dưới và ngưỡng trên của tập tìm kiếm, còn lại để tìm kiếm nhị phân ta cần kiểm tra xem khi đem thử 1 kích thước cho đồng xu nằm ở giữa là R, xem nó $>$ hoặc $<$ hoặc $=$ rKQ

Kích thước đem thử là R, ta sẽ tính số đo của mỗi N góc mà góc thứ i là góc tạo bởi 2 tiếp tuyến kẻ từ tâm đồng xu nằm giữa đến đồng xu thứ i:



Hình tròn trên là đồng xu thứ i. Để ý tứ giác màu đỏ, nó có 2 góc vuông đối nhau, 2 cạnh nhỏ bằng nhau và bằng $r[i]$, đường chéo (tô màu xanh lá cây) $= (r[i] + R)$, nhờ các hàm lượng giác tính được góc tô màu xanh da trời và một tiếp tuyến là $\sin^{-1} \frac{r[i]}{r[i] + R}$, từ đó suy ra góc giữa hai tiếp tuyến đó là $2 \sin^{-1} \frac{r[i]}{r[i] + R}$

Mặt khác, ta cũng có nhận xét: R tăng $\Leftrightarrow \frac{r[i]}{r[i] + R}$ giảm $\Leftrightarrow 2 \sin^{-1} \frac{r[i]}{r[i] + R}$ giảm (góc nhọn) \Leftrightarrow tổng N góc đó giảm.

Gọi tổng N góc đó là SumAngle:

- Nếu SumAngle = 360 độ \Leftrightarrow OK \Leftrightarrow rKQ = R và kết thúc tìm kiếm!

- Nếu $\text{SumAngle} < 360^\circ \Leftrightarrow rKQ < R \Rightarrow$ tìm kiếm tiếp ở khoảng dưới.
- Nếu $\text{SumAngle} > 360^\circ \Leftrightarrow rKQ > R \Rightarrow$ tìm kiếm tiếp ở khoảng trên.

Một lưu ý chung cho các bài toán hình học là: vì máy tính xử lý số thực 1 cách xấp xỉ nên ta cần 1 hằng số epsilon cho việc so sánh $> < =$ các số thực.

Sau đây là chương trình trên ngôn ngữ Pascal:

```

Program ChuongTrinh;
uses crt;
const Epsilon = 0.0001;
var r: array [1..100] of real;
    RR: real;
    n: integer;
    CanDuoi, CanTren: real;

procedure Nhap;
var i: integer;
begin
    write('Nhap so luong hinh tron: '); readln(n);
    for i:= 1 to n do
    begin
        write('    Nhap ban kinh hinh tron thu ', i,
' : ');
        readln(r[i]);
    end;
end;

procedure XacDinhTiemCan;
var i: integer;
begin
    CanDuoi:= 0;
    CanTren:= 0;
    for i:= 1 to n do CanTren:= CanTren + r[i];
    CanTren:= CanTren*2;
end;

function arcsin(x: real): real;
begin
    arcsin:= arctan(x/sqrt(1-x*x));
end;

function Tong_N_Goc(RR: real):real;

```

```
var Sum: real;
    i: integer;
begin
    Sum:= 0;
    for i:= 1 to n do Sum:= Sum +
2*arcsin(r[i]/(r[i] + RR));
    Tong_N_Goc:= Sum;
end;

procedure TimKiemNhiPhan;
var Found: boolean;
    Sum: real;
begin
    Found:= false;
    while not Found do
    begin
        RR:= (CanDuoi + CanTren)/2;
        Sum:= Tong_N_Goc(RR);

        if Sum < 2*pi then CanTren:= RR
            else CanDuoi:= RR;

        if (CanTren - CanDuoi) < Epsilon then
        begin
            Found:= true;
            RR:= CanTren;
        end;
    end;
    writeln('Ban kinh can tim la: ', RR:0:4);
end;

Begin { Chuong Trinh Chinh }
    clrscr;
    Nhap;
    XacDinhTiemCan;
    TimKiemNhiPhan;
    readln;
End.
```