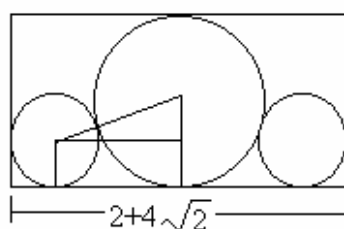


算法实现题 7-9 圆排列问题

★问题描述:

给定 n 个大小不等的圆 c_1, c_2, \dots, c_n ，现要将这 n 个圆排进一个矩形框中，且要求各圆与矩形框的底边相切。圆排列问题要求从 n 个圆的所有排列中找出有最小长度的圆排列。例如，当 $n=3$ ，且所给的 3 个圆的半径分别为 1, 1, 2 时，这 3 个圆的最小长度的圆排列如图所示。其最小长度为 $2+4\sqrt{2}$ 。



解圆排列问题的一个随机化算法如下。

```
void Circle_search(int *x)
{
    random_perm(x);
    found=true;
    while(found){
        found=false;
        for(int i=1;i<=n;i++)
            for(int j=1;j<=n;j++)
                if(swap(x[i],x[j]) reduces length){
                    swap(x[i],x[j]);
                    found=true;}
    }
}
```

其中，**random_perm(x)**产生 x 的一个随机排列。

★编程任务:

根据上述算法框架，设计一个随机化算法，对于给定的 n 个圆，计算 n 个圆的最佳排列方案，使其长度尽可能小。

★数据输入:

由文件 input.txt 给出输入数据。第一行有 1 个正整数 n ($1 \leq n \leq 20$)。接下来的 1 行有 n

个数，表示 n 个圆的半径。

★结果输出:

将计算出的最小圆排列的长度输出到文件 `output.txt`。

输入文件示例

`input.txt`

3

1 1 2

输出文件示例

`output.txt`

7.65685