**打印十字图**

问题描述

小明为某机构设计了一个十字型的徽标（并非红十字会啊），如下所示：

..$$$$$$$$$$$$$..  
..$...........$..  
$$$.$$$$$$$$$.$$$  
$...$.......$...$  
$.$$$.$$$$$.$$$.$  
$.$...$...$...$.$  
$.$.$$$.$.$$$.$.$  
$.$.$...$...$.$.$  
$.$.$.$$$$$.$.$.$  
$.$.$...$...$.$.$  
$.$.$$$.$.$$$.$.$  
$.$...$...$...$.$  
$.$$$.$$$$$.$$$.$  
$...$.......$...$  
$$$.$$$$$$$$$.$$$  
..$...........$..  
..$$$$$$$$$$$$$..

对方同时也需要在电脑dos窗口中以字符的形式输出该标志，并能任意控制层数。

输入格式

一个正整数 n (n<30) 表示要求打印图形的层数。

输出格式

对应包围层数的该标志。

样例输入1

1

样例输出1

..$$$$$..  
..$...$..  
$$$.$.$$$  
$...$...$  
$.$$$$$.$  
$...$...$  
$$$.$.$$$  
..$...$..  
..$$$$$..

样例输入2

3

样例输出2

..$$$$$$$$$$$$$..  
..$...........$..  
$$$.$$$$$$$$$.$$$  
$...$.......$...$  
$.$$$.$$$$$.$$$.$  
$.$...$...$...$.$  
$.$.$$$.$.$$$.$.$  
$.$.$...$...$.$.$  
$.$.$.$$$$$.$.$.$  
$.$.$...$...$.$.$  
$.$.$$$.$.$$$.$.$  
$.$...$...$...$.$  
$.$$$.$$$$$.$$$.$  
$...$.......$...$  
$$$.$$$$$$$$$.$$$  
..$...........$..  
..$$$$$$$$$$$$$..

提示

请仔细观察样例，尤其要注意句点的数量和输出位置。

锦囊1

使用二维数组。

本题的C++参考代码如下：

#include <stdio.h>

#include <malloc.h>

#include <string.h>

int main()

{

int n,w,h,l,i,j,x,y,m;

char \*arry;

scanf("%d",&n);

w=h=5+n\*4;

arry=(char \*)malloc(w\*h);

memset(arry,'.',w\*h);

/\*

for(i=n\*2;i<w-n\*2;i++)//画基础图形

{

arry[i+((n+1)\*2)\*w]='$';

arry[i\*w+(n+1)\*2]='$';

}\*/

for(m=0;m<=n;m++)//画没一层

{

for(i=(m+1)\*2;i<w-(m+1)\*2;i++)//四边

{

x=m\*2;

y=i;

arry[x+y\*w]='$';

x=w-m\*2-1;

y=i;

arry[x+y\*w]='$';

x=i;

y=m\*2;

arry[x+y\*w]='$';

x=i;

y=w-m\*2-1;

arry[x+y\*w]='$';

}

for(i=m\*2;i<=(m+1)\*2;i++)//角

{

x=i;

y=(m+1)\*2;

arry[x+y\*w]='$';

x=(m+1)\*2;

y=i;

arry[x+y\*w]='$';

x=w-i-1;

y=(m+1)\*2;

arry[x+y\*w]='$';

x=w-(m+1)\*2-1;

y=i;

arry[x+y\*w]='$';

x=i;

y=h-(m+1)\*2-1;

arry[x+y\*w]='$';

x=(m+1)\*2;

y=h-i-1;

arry[x+y\*w]='$';

x=w-i-1;

y=h-(m+1)\*2-1;

arry[x+y\*w]='$';

x=w-(m+1)\*2-1;

y=h-i-1;

arry[x+y\*w]='$';

}

}

for(i=0;i<h;i++)

{

for(j=0;j<w;j++)

{

printf("%c",arry[i\*w+j]);

}

printf("\n");

}

return 0;

}

本题的C参考代码如下：

#include <stdio.h>

#include <malloc.h>

#include <string.h>

int main()

{

int n,w,h,l,i,j,x,y,m;

char \*arry;

scanf("%d",&n);

w=h=5+n\*4;

arry=(char \*)malloc(w\*h);

memset(arry,'.',w\*h);

/\*

for(i=n\*2;i<w-n\*2;i++)//画基础图形

{

arry[i+((n+1)\*2)\*w]='$';

arry[i\*w+(n+1)\*2]='$';

}\*/

for(m=0;m<=n;m++)//画没一层

{

for(i=(m+1)\*2;i<w-(m+1)\*2;i++)//四边

{

x=m\*2;

y=i;

arry[x+y\*w]='$';

x=w-m\*2-1;

y=i;

arry[x+y\*w]='$';

x=i;

y=m\*2;

arry[x+y\*w]='$';

x=i;

y=w-m\*2-1;

arry[x+y\*w]='$';

}

for(i=m\*2;i<=(m+1)\*2;i++)//角

{

x=i;

y=(m+1)\*2;

arry[x+y\*w]='$';

x=(m+1)\*2;

y=i;

arry[x+y\*w]='$';

x=w-i-1;

y=(m+1)\*2;

arry[x+y\*w]='$';

x=w-(m+1)\*2-1;

y=i;

arry[x+y\*w]='$';

x=i;

y=h-(m+1)\*2-1;

arry[x+y\*w]='$';

x=(m+1)\*2;

y=h-i-1;

arry[x+y\*w]='$';

x=w-i-1;

y=h-(m+1)\*2-1;

arry[x+y\*w]='$';

x=w-(m+1)\*2-1;

y=h-i-1;

arry[x+y\*w]='$';

}

}

for(i=0;i<h;i++)

{

for(j=0;j<w;j++)

{

printf("%c",arry[i\*w+j]);

}

printf("\n");

}

return 0;

}

本题的Java参考代码如下：

import java.util.Scanner;

public class Main{

public static void main(String[] args){

Scanner sc=new Scanner(System.in);

fd(sc.nextInt());

}

public static void fd(int n){

int len=5+4\*n;

String a[][]=new String[len][len];

int b=len/2;

for(int i=0;i<len;i++){

for(int j=0;j<len;j++){

a[i][j]=".";

}}

for(int i=b-2;i<b+2;i++){

a[i][b]="$";

a[b][i]="$";

}

for(int m=0;m<n;m++){

for(int i=b-2-2\*m;i<(b+3+2\*m);i++){

a[b - (4+m\*2)][i] = "$";

a[b + (4+m\*2)][i] = "$";

a[i][b - (4+m\*2)] = "$";

a[i][b + (4+m\*2)] = "$";

}

}

for(int m=0;m<n;m++){

for (int i = b - (3+m\*2); i <= b + 3+m\*2; i++) {

a[b - (2+m\*2)][i] = "$";

a[b + 2+m\*2][i] = "$";

a[i][b + 2+m\*2] = "$";

a[i][b - (2+m\*2)]= "$";

}

a[b - (2+m\*2)][b - (1+m\*2)] = ".";

a[b - (2+m\*2)][b + 1+m\*2] = ".";

a[b + 2+m\*2][b - (1+m\*2)] = ".";

a[b + 2+m\*2][b + 1+m\*2] = ".";

a[b - (1+m\*2)][b - (2+m\*2)] = ".";

a[b + 1+m\*2][b - (2+m\*2)] = ".";

a[b - (1+m\*2)][b + 2+m\*2] = ".";

a[b + 1+m\*2][b + 2+m\*2] = ".";

}

for (int i= 0; i < len; i++) {

for (int j = 0; j < len; j++) {

System.out.print(a[i][j]);

}

System.out.println();

}

}

}