**算法训练 输出米字形**

时间限制：1.0s   内存限制：512.0MB

　例如:n=3时，包含A,B,C；n=4时，包含A,B,C,D。  
　　矩阵的正中间为n个字母中字典序最大的那个，从这个字母开始，沿着西北、正北、东北、正西、正东、西南、正南、东南八个方向各有一条由大写字母组成的直线。并且直线上的字母按字典序依次减小，直到大写字母A。  
　　矩阵的其它位置用英文句号．填充。  
  
　　样例输入一  
　　3  
  
　　样例输出一  
　　Ａ．Ａ．Ａ  
　　．ＢＢＢ．  
　　ＡＢＣＢＡ  
　　．ＢＢＢ．  
　　Ａ．Ａ．Ａ  
  
　　样例输入二  
　　4  
  
　　样例输出二  
　　Ａ．．Ａ．．Ａ  
　　．Ｂ．Ｂ．Ｂ．  
　　．．ＣＣＣ．．  
　　ＡＢＣＤＣＢＡ  
　　．．ＣＣＣ．．  
　　．Ｂ．Ｂ．Ｂ．  
　　Ａ．．Ａ．．Ａ

本题的C++参考代码如下：

#include "iostream"

using namespace std;

const int MAXN = 10000;

char g[MAXN][MAXN];

void init(int n)

{

for (int i = 0; i < 2\*n-1; ++i)

{

for (int j = 0; j < 2\*n-1; ++j)

{

g[i][j] = '.';

}

}

}

void fillg(int n)

{

int m = (2\*n-1)/2;

for (int i = 0; i <= m; ++i)

{

g[i][i] = 'A'+i;

g[m][i] = 'A'+i;

g[i][m] = 'A'+i;

}

for (int i = 0; i <= m; ++i)

{

for (int j = 2\*n-2; j > m; --j)

{

g[i][j] = g[i][2\*n-2-j];

}

}

for (int i = 2\*n-2; i > m; --i)

{

for (int j = 0; j<2\*n-1; ++j)

{

g[i][j] = g[2\*n-2-i][j];

}

}

}

int main()

{

int n;

cin>>n;

init(n);

fillg(n);

for (int i = 0; i < 2\*n-1; ++i)

{

for (int j = 0; j < 2\*n-1; ++j)

{

cout<<g[i][j];

}

cout<<endl;

}

cout<<endl;

return 0;

}

本题的C参考代码如下：

#include <stdio.h>

int main()

{

int n,i,j,k,t;

scanf("%d",&n);

char a[2\*n-1][2\*n-1];

for(i=0;i<2\*n-1;i++)

for(j=0;j<2\*n-1;j++)

a[i][j]=0;

k=n;

for(i=0;i<n-1;i++) //行数

{

k--; //k为字符间隔

t=i;

for(j=1;j<=3;j++)

{

a[i][t]='A'+i;

t+=k;

}

}

for(i=0;i<n;i++) //中间一行

{

a[n-1][i]='A'+i;

a[n-1][2\*n-2-i]='A'+i;

}

for(i=0;i<=n-2;i++) //填补句号

for(j=0;j<2\*n-1;j++)

{

if(a[i][j]==0)

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

}

for(i=0;i<=n-2;i++) //翻转

for(j=0;j<2\*n-1;j++)

a[2\*n-2-i][j]=a[i][j];

for(i=0;i<2\*n-1;i++)

{

for(j=0;j<2\*n-1;j++)

printf("%c",a[i][j]);

printf("\n");

}

return 0;

}

本题的Java参考代码如下：

import java.io.BufferedReader;

import java.io.IOException;

import java.io.InputStreamReader;

public class Main{

public static void main(String[] args) throws IOException {

BufferedReader br = new BufferedReader(new InputStreamReader(System.in));

int n = Integer.parseInt(br.readLine());

String str = "";

for (int i = 0; i < n; i++)

str += (char) (i + 65);

for (int i = n - 2; i > -1; i--)

str += (char) (i + 65);

int len = n \* 2 - 1;

char[][] chs = new char[len][len];

int y = len - 1;

for (int i = 0; i < chs.length; i++) {

chs[i][i] = str.charAt(i);

chs[i][y--] = str.charAt(i);

chs[len / 2][i] = str.charAt(i);

chs[i][len / 2] = str.charAt(i);

}

for (int i = 0; i < chs.length; i++) {

for (int j = 0; j < chs[i].length; j++) {

if(!(chs[i][j]>='A' && chs[i][j]<='Z'))

chs[i][j] = '.';

}

}

for (int i = 0; i < chs.length; i++) {

for (int j = 0; j < chs[i].length; j++) {

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

}

System.out.println();

}

}

}