#include <stdio.h>

void Hanoi(int n,char a,char b,char c);

void Move(int n,char a,char b);

int main()

{

int n;

printf("Input the number of disks:");

scanf("%d",&n);

printf("Steps of moving %d disks from A to B by means of C:\n",n);

Hanoi(n,'A','B','C');

return 0;

}

void Hanoi(int n,char a,char b,char c)

{

if(n==1)

{

Move(n,a,b);

}

else

{

Hanoi(n-1,a,c,b);

Move(n,a,b);

Hanoi(n-1,c,b,a);

}

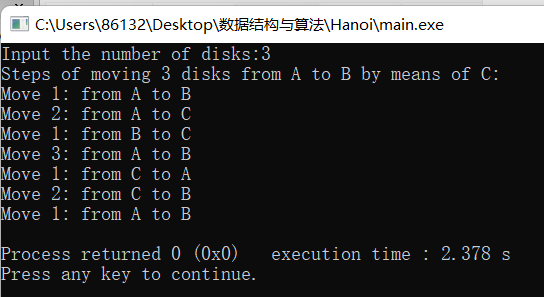
}

void Move(int n,char a,char b)

{

printf("Move %d: from %c to %c \n",n,a,b);

}



设盘子个数为n时，需要T（n)步，把A柱子n-1个盘子移到B柱子，需要T（n-1)步，A柱子最后一个盘子移到C柱子一步，B柱子上n-1个盘子移到C柱子上T（n-1)步。  
得递推公式T（n）=2T（n-1)+1  
所以汉诺塔问题的时间复杂度为O(2^n);

空间复杂度为O(1);