visual studio调试

F9或者代码左边点击一下，出现红点，设置断点

F10执行下一行代码

F11进入当前代码所调用的方法，如果没有则进行下一句

F5跳过本断点。跳到下一个断点

调试——删除所有断点

#include"stdafx.h"

#include<stdio.h>

int n, a[10], book[10];//特别说明c语言全局变量没有赋值默认为0,无需再次初始化；

void dfs(int step)//step 表示当前在第几个位置

{

int i;

if (step == n + 1)//如果step==n+1表示前n个数字已经放好

{

//输出一种全排列

for (i = 1; i <= n; i++)

printf("%d", a[i]);

printf("\n");

return;

}

//每次搜索都从1-n 一一尝试

for (i = 1; i <= n; i++)

{

if (book[i] == 0)//判断次数字是否用过

{

a[step] = i;//存储当前位置的数字，以便满足条件输出

book[i] = 1;//当前数字已用过，改变标志，以防重用

dfs(step + 1);//放好当前位置数字之后，安排下一个数字

book[i] = 0;//回溯，当满足一种全排列后，进行下一种尝试

}

}

return;

}

int main()

{

scanf\_s("%d", &n);//输出1-n的全排序（n<=9)

dfs(1);//从第一个位置开始

return 0;

}

深度优先产生全排列

for(i=1;i<=n;i++)

{

dfs(step+1);

}

核心思想，for是对每个节点的所有子节点，i=0时再次dfs——for，就是深度优先