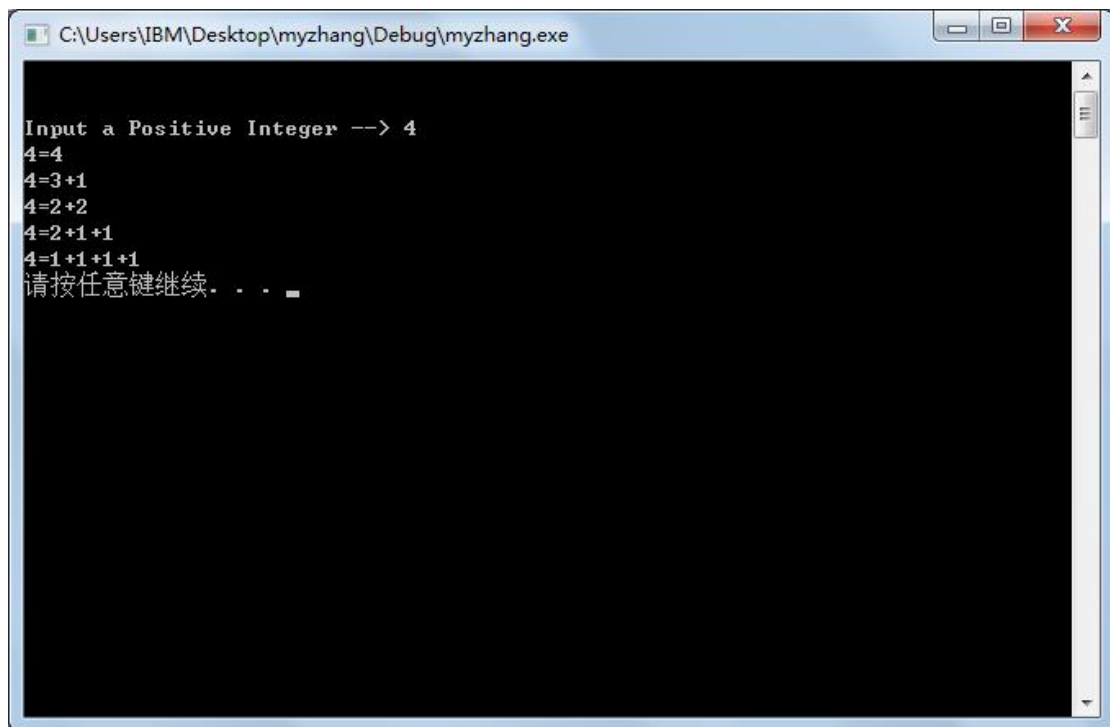


/\*编程：对给定的  $n$ , 输出其和等于  $n$  的所有不增的正整数和式的函数。例如：  $n=4$ ，程序将输出：



```
C:\Users\IBM\Desktop\myzhang\Debug\myzhang.exe

Input a Positive Integer --> 4
4=4
4=3+1
4=2+2
4=2+1+1
4=1+1+1+1
请按任意键继续. . .
```

提示：可以用递归或非递归解题。

引入全局数组  $a[]$  用于存储分解出来的和数， $a[k]$  存储第  $k$  步分解出来的和数。若用递归，则递归函数设置两个参数，参数  $i$  是本次递归调用要分解的数，参数  $k$  是本次递归调用将分解出第  $k$  个和数。对要分解的  $i$ ，可以分解的数  $j$  共有  $i$  种可能的选择， $i, i-1, \dots, 2, 1$ 。但为保证分解出来的和数依次构成不增的正整数序列，要求从  $i$  分解出来的和数  $j$  不能超过  $a[k-1]$ 。为保证上面要求对第一步 ( $k=1$ ) 分解也成立，程序可在  $a[0]$  预置  $n$ ，即第一个和数最大为  $n$ 。在分解的过程中，当分解的出来的数  $j=i$  时，说明已经完成了一个完整的和式分解，应该将和式输出，否则当分解出来的数  $j < i$  时，说明还有  $i-j$  需要分

解出第  $k+1$  个和数。

```
#include <stdio.h>
```

```
#include <stdlib.h>
```

```
#define MAXSIZE 50
```

```
int a[MAXSIZE];
```

```
void partition(int i,int k)
```

```
{
```

请在此编程

```
}
```

```
void main()
```

```
{
```

```
int n;
```

```
printf("\n\nInput a Positive Integer --> ");
```

```
scanf("%d",&n);
```

```
a[0]=n;
```

```
partition(n,1);
```

```
system("pause");
```

```
}
```

