##### 递归算法

定义数组Num[120]，Num[0]存储输入的数N，随后的数组元素存储拆分的自然数，使用变量k表示拆分位数，例如7＝1＋6，则Num[0]＝7，Num[1]＝1，Num[2]＝6，k＝2。继续拆分从Num[k]开始，Num[k]能否再拆分取决于Num[k]/2是否大于等于Num[k－1]。

参考代码如下所示。

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39 | //拆分自然数 — 递归算法  #include <bits/stdc++.h>  using namespace std;  int Num[120],Count;  void Print(int k) //打印k个数  {  printf("%d=%d",Num[0],Num[1]);  for(int i=2; i<=k; i++)  printf("+%d",Num[i]);  printf("\n");  ++Count;  }  void Split(int k,int n) //k为指向的位数,n为待划分的数  {  if(n<=0) //如无法再拆分，则打印  Print(k);  else  for(int i=Num[k]; i<=n; ++i) //后面的数大于等于前面的数  {  Num[k+1]=i; //则成功拆分  Split(k+1,n-Num[k+1]); //并进入下一个数的递归  }  }  int main()  {  scanf("%d",&Num[0]); //例如输入7  for(int i=1; i<=Num[0]/2; ++i) //先拆分成两位加数  {  Num[1]=i; //如Num[0]为7，Num[1]只可能是1,2,3  Split(1,Num[0]-Num[1]); //递归继续拆分  }  printf("%d\n",Count);  return 0;  } |