整数因子分解问题

张家强　　　　SC17023010

2018年4月1日

1　题目介绍

1.1　问题描述

大于1的正整数 n 都可以分解为 $n = x_1 * x_2 * ... * x_m$.
例如：当n=12时，共有8种不同的分解式： 12 = 12
12 = 6*2
12 = 4*3
12 = 3*4
12 = 3*2*2
12 = 2*6
12 = 2*3*2
12 = 2*2*3

1.2　算法要求

对于给定正整数n，计算n共有多少种不同的分解式。

1.3　数据输入输出

由文件input.txt给出输入数据。第一行有一个正整数$n(1 \leq n \leq 2000000000)$。将计算出的不同的分解式数输出到文件output.txt。

2　题目分析

此题因子讲顺序的.第一个因子可能是2 n之间的数。比如对12而言,第一个因子可能是2,3,4,6,12。

将第一个因子为2的分解个数,加上第一个因子为3的分解个数等等，一直至加到第一个因子为12的分解个数。而第一个因子为2的分解个数为6(因为12/2=6)的分解个数,所以递归即可求解。本题可利用两种递归方法求解，一种为完全纯递归方法，一种动态规划中的备忘录方法。

3 算法设计

3.1 纯递归方法

纯递归方法的算法设计较为简单。当输入为1时，直接返回1。若输入不为1，则从2开始到n进行递归调用，每当用一个因子能整除n，计数加一。为了减少循环次数和递归调用次数，结合质数筛选定理:“n不能够被不大于根号n的任何质数整除，则n是一个质数”，将函数内的循环上限设定为$\sqrt{n}$。递归函数的传入值为由文件输入的正整数n，内部有一初值为1的计数变量cnt，记录n的分解式个数。函数首先判断n是否为1，若为1，则返回1；若不为1，则开始函数内从2到$\sqrt{n}$自增1的循环，当i能整除n时，判断i是否等于$\sqrt{n}$，若相等则cnt等于cnt加solve(i)的返回值，否则$cnt = cnt + solve(n/i) + solve(i)$。最终函数返回cnt。代码如下：

```
int solve(int n) {
        if (n == 1) return 1;
                int cnt = 1;
        for (int i = 2; i <= sqrt(n); i++) {
                if (n % i == 0) {
                        if (i == sqrt(n))
                                cnt += solve(i);
                        else
                                cnt += solve(n / i) + solve(i);
                }
        }
        return cnt;
}
```

3.2 备忘录方法

在查阅相关问题后得知，本题还可以利用动态规划中的备忘录方法求解。备忘录方法是动态规划算法的变形。与动态规划算法一样，备忘录方法用表格保存已解决的子问题的答案，

在下次需要解此子问题时，只要简单地查看该子问题的解答，而不必重新计算。

此方法先定义全局int型指针num和全局int型变量length，分别保存因子的分解式个数和num数组的长度。计算方法与纯递归方法相同，不同的是在每次计算后将cnt保存到num中。代码如下：

```
int* num;
int length;
int lookup(int n) {
        if (n < length && num[n] != 0)  return num[n];
        int cnt = 1;
        int q = sqrt(n);
        for (int i = q; i >= 2; i--) {
                if (n % i == 0) {
                        if (i == sqrt(n)) cnt += lookup(i);
                        else cnt += lookup(i) + lookup(n / i);
                }
        }
        if (n < length) num[n] = cnt;
        return cnt;
}
```

4 算法分析

4.1 递归式

$$f(n)=\begin{cases}1, & \text{n=1}\\ \sum_{n\%i=0,i\neq n/i}(f(i)+f(n/i))+\sum_{n\%i=0,i=n/i}f(i), & n\geq 2\end{cases}$$

4.2 复杂度

本题的复杂度取决于n的大小和n因子的多少，当n趋向于无穷大时，求其因子和因子数量的算法复杂度还不能确定。