



***Liaoning Normal University***

**算法设计分析实验报告**

**课程名称：计算机算法设计分析**

**专业：计算机科学与技术（师范）**

**班级： （1） 班**

**学号：201621012299**

**姓名：周惠馨**

**指导教师：王相海**

**2018年12月13日**

1. **问题的提出**

最优分解问题。设n是一个正整数。现在要求将n分解为若干互不相同的自然数的和，且使这些自然数的乘积最大。

算法设计：对于给定的正整数n，计算最优分解方案。

数据输入：有文件input.txt提供输入数据。文件的第一行是正整数n。

结果输出：将计算的最大乘积输出到文件output.txt。

1. **算法的基本思想**

贪心算法（又称贪婪算法）是指，在对问题求解时，总是做出在当前看来是最好的选择。也就是说，不从整体最优上加以考虑，他所做出的是在某种意义上的局部最优解。

贪心算法不是对所有问题都能得到整体最优解，关键是贪心策略的选择，选择的贪心策略必须具备无后效性，即某个状态以前的过程不会影响以后的状态，只与当前状态有关。

1. **算法的程序实现**

#include <stdio.h>

int fun(int n){

int a[100];//保存分解后的数

int k=1,sum=1;//k为数组a索引，sum乘积

int i,j;

a[1]=2; //先减2,注意这里索引是1

n-=2;

while(n>a[k]){ //循环减

k++;

a[k]=a[k-1]+1;

n-=a[k];

}

if(n==a[k]){//相等，最后的再加1。

a[k]++;

n--;

}

for(i=0;i<n;i++)//以10为例，这里n=1，然后每个元素+1

a[k-i]++;

for(j=1;j<=k;j++){//从1开始

sum\*=a[j];

}

return sum;

}

int main(){

int n,m;

FILE \*fin,\*fout;

fin = fopen("input.txt","r"); // 打开文件，按读的方式打开

fscanf(fin,"%d", &n);

printf("input.txt输入n:10\n");

m=fun(n);

fout=fopen("output.txt", "w");// 打开文件，按写的方式打开

fprintf(fout,"%d",m);

fclose(fout);

printf("输出%d output.txt\n",m);

return 0;

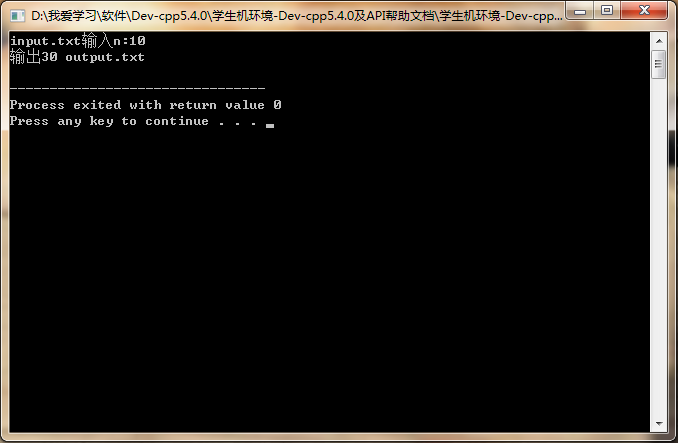
}

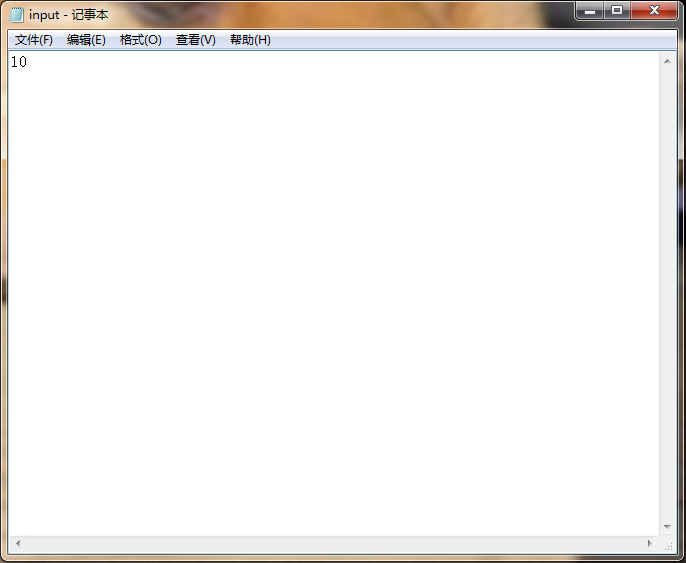
1. **算法的复杂度分析**

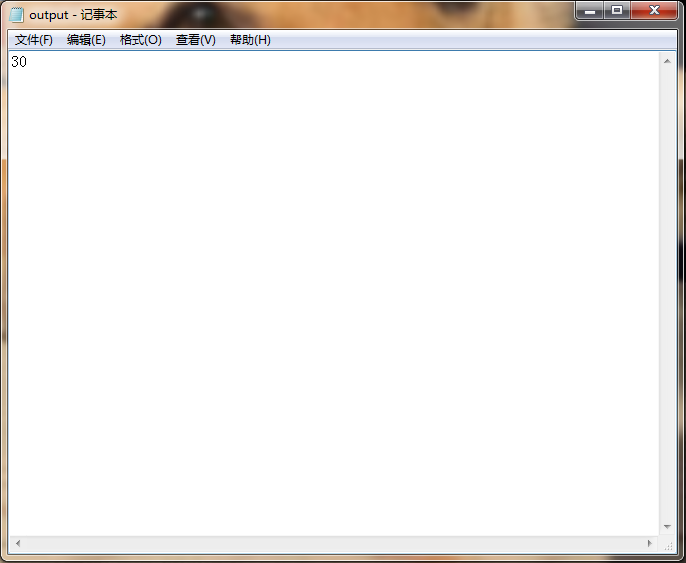
空间复杂度：O(n^2)

时间复杂度：O(n^2)

1. **算法运行结果与分析（拷屏黏贴）**







1. **结束语（讨论）**

贪心算法是从问题的某个初始解出发逐步，逼近给定的目标，以尽可能快地求得更好的解。当达到某一步不能继续前进时，算法停止。这时就得到了问题的一个解。但不能保证求得的解是最优的。贪心算法的优点在于时间复杂度低。贪心算法与其他最优算法的区别在于：它具有不可后撤性，可以有后效性，一般情况下，不能满足最优化原理。贪心算法的特点就决定了它的使用范围，它一般不适用于解决可行性问题。仅适用于较容易得到可行性解得最优性问题。这里较容易得到可行解得概念：当前的策略选择后，不会或极少出现无解的情况。交互性题目，贪心算法是一个较好的选择。