第三章算法分析题

学号：2209060322　　姓名：梁桐　　班级：计算机 2203

第三章 算法分析题

梁桐 计算机2203 2209060322

3-2解

(1)问题设定与假设：设已知数组a[0:i-1]的最长递增子序列的长度，并希望引入新元素a[i]来扩展该序列。设数组长度为n，最终需要得到长度为n的数组a[0:n-1]的最长递增子序列的长度

(2)递推关系：设P是序列a[0:i-1]中的最长递增子序列的长度，且P的最后一个元素为a[j]，其中0≤j<i，在判断a[i]是否可以扩展该序列时，分以下几种情况：

若a[j]<a[i]，可将a[i]添加至序列末尾，此时最长递增子序列长度加1

若a[j]≥a[i]，则无法扩展该序列，考虑其它可能性.

(3)辅助数组的使用：使用辅助数组b，其中b[k]表示长度为k+1的递增子序列的最大结尾元素。对于每一个新元素a[i]通过与b中的元素进行比较，来确定它是否能扩展某些序列的长度

若a[i]>b[k]，则a[i]可以作为新长度为k+2的递增序列的结尾元素，此时更新b[k+1]=a[i]

若a[i]≤b[k]，则寻找合适的j使得b[j]<a[i]<b[j+1]，并更新b[j+1]=a[i]以保持递增序列的最小结尾

(4)算法优化：在每次循环中更新b数组通过使用二分查找加快寻找适合插入的位置，这样可以保证算法的时间复杂度为O(nlogn)

实现如下：

```
int LIS() {
  b[1] = a[0];
  for(int i=1, k=1; i<n; i++){
    if(a[i]>=b[k])
       b[++k] = a[i];
    else
```

```
        b[binary(i,k)]=a[i];
    }
    return k;
}
int binary(int i, int k){
    if(a[i] < b[1])
        return 1;
    for(int h=1, j=k, h!=j-1;){
        if(b[k=(h+j)/2] <= a[i])
            h=k;
        else
            j=k;
    }
    return j;
}
```

其中，binary(i, k)用二分搜索算法在b[1:k]中找到下标j，使得$b[j-1] \le a[i] < b[j]$。算法binary(i,k)所需时间为$O(\log k)$。符合题意。

3-3 解答

设所给背包问题的子问题 $\max\sum_{k=1}^{i} c_k x_k$ $\sum_{k=1}^{i} a_k x_k \le j$ 的最优值为$m(i,j)$，即$m(i,j)$是背包容量为j，可选择物品为$1,2,\cdots,i$时背包问题的最优值。由背包问题的最优子结构性质，可建立计算$m(i,j)$的递归式如下。

$$m(i,j)=\begin{cases}\max\{m(i-1,j),\ m(i,j-a_i)+c_i\} & a_i\le j\\ m(i-1,j) & 0\le j<a_i\end{cases}$$

$m(0,j)=m(i,0)=0$；$m(i,j)=-\infty$，$j<0$

按此递归式计算的$m(n,b)$为最优值，算法所需计算时间为$O(nb)$