数据结构与算法实验三

求最长单调递增子序列

2011013251 软件 11 吕婉琪

目录

实验目的

给出一个 O(nlgn)的算法，使之能够求一个长度为 n 的序列的最长单调递增子序列，有方便调试的图形界面。

算法思路

定义原始数组为 a[1],a[2],...,a[n]，定义 S[i]为以 a[i]结尾的最长递增子序列的长度，则要使：s[i]=max{s[r]|1 = <r < i, a[i] > a[r]}+ 1。初始情况为 s[1]=1。插入位置可用二分查找获得。伪代码如下：

```
largesub(X, n)
L = 0
for i = 1, 2, … n
binary search for the largest positive j ≤ L such that X[M[j]] < X[i]
(or set j = 0 if no such value exists)
P[i] = M[j]
if j == L or X[i] < X[M[j+1]]
M[j+1] = i
L = max(L, j+1)
```

运行结果

运行环境

OS: Windows 8 专业版 64位

CPU: Intel® CoreTM i7-2720QM CPU @2.20GHz 2.20GHz RAM: 4.00GB

开发工具: Microsoft Visual Studio 2012

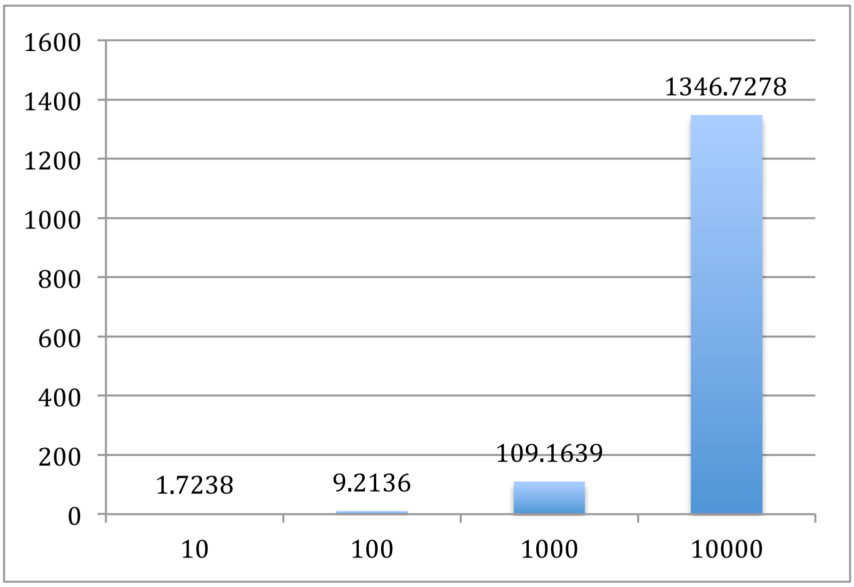
语言：Visual C#

编译器选项: Release

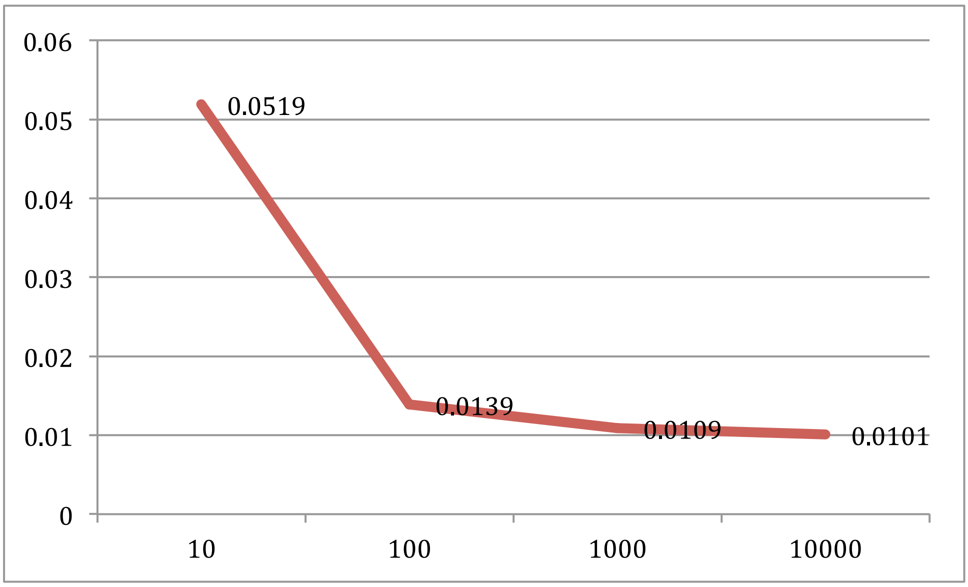
备注:可执行文件hw4.exe在根目录下

时间对比(ns)及算法复杂度

不同数据量的运行时间对比：



不同数据量下，T(n)/nlgn 的值：



可以看到，除第一组规模为 10 的序列外，其他 T(n)/nlgn 的值基本相同。故可以认为算法的时间复杂度是 O(nlgn)。

(备注：由于我是用 split 函数将输入的字符串分成字符串数字序列的，故输入只能分辨单个空格间隔的序列)