

《数据结构》上机实验报告

第 1 次上机

学号：

姓名： yyh

学院： 信息科学学院

专业： 计算机科学

教师：

日期： 2018.9.12

1. 实验要求

1.上机之前应做好充分准备，认真思考所需的上机题目，提高上机效率。

2.独立上机输入和调试自己所编的程序，切忌抄袭、拷贝他人程序。

3.上机结束后，整理出实验报告。书写报告时，重点放在实验的方法、思路以及总结反思上，以达到巩固课堂学习、提高动手能力的目的。

1. 实验内容

给定(可负)整数A1，A2，…，AN，求Σjk=i Ak的最大值。（为了方便，如果所有整数都是负数，则规定最大子列之和为0。）

例如：输入-2, 11, -4, 13, -5, -2的答案是20 (A2到A4)。

1. 实验步骤（写出问题分析或者算法思路）

思路：

因为是连续的子序列，考虑使用前缀和来化简所求的式子。

1定义前缀和

2利用前缀和来简化所求式子：

3 代入Max并化简

实现：

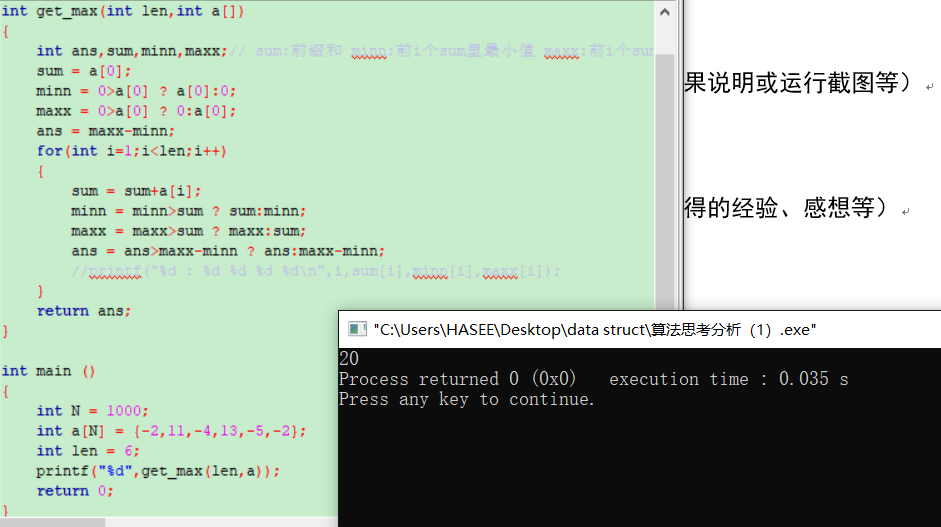
维护一个前i个的最大值和前i个的最小值。用代码里的符号表示就是 **和 。**

**但是和的更新不需要从1到i地去遍历，只需要让sum[i]和之前的最大（最小）值比较就好了：**

最后*i*按照从1到n顺序递推，并更新答案即可。

时间复杂度和空间复杂度都是O(n)。

1. 程序清单（源程序代码等）
2. #include<stdio.h>
3. //最大子序列求和问题 复杂度O(n)
4. **int** get\_max(**int** len,**int** a[])
5. {
6. **int** ans,sum,minn,maxx;// sum:前缀和 minn:前i个sum里最小值 maxx:前i个sum里最大值
7. sum = a[0];
8. minn = 0>a[0] ? a[0]:0;
9. maxx = 0>a[0] ? 0:a[0];
10. ans = maxx-minn;
11. **for**(**int** i=1;i<len;i++)
12. {
13. sum = sum+a[i];
14. minn = minn>sum ? sum:minn;
15. maxx = maxx>sum ? maxx:sum;
16. ans = ans>maxx-minn ? ans:maxx-minn;
17. }
18. **return** ans;
19. }
21. **int** main ()
22. {
23. **int** N = 1000;
24. **int** a[N] = {-2,11,-4,13,-5,-2};
25. **int** len = 6;
26. printf("%d",get\_max(len,a));
27. **return** 0;
28. }
29. 运行结果（程序运行时的结果说明或运行截图等）



1. 总结（实验中遇到的问题、取得的经验、感想等）

1 对原式进行化简的时候用到了部分数学上的公式。

2 实现的时候还考虑过使用新的数组记录max和min值，但是后来发现并不需要，从而进一步减少了内存空间的使用。（主要是因为最后求答案时，只需要用到max[len]和min[len]。但如果还要求输出起点和终点的话就需要用数组记录了，方便查找。）

3 感觉应该多想再写，多想能少改，更节约时间。

4 把a[n]定义成全局变量能更加减少对空间的使用（O(1)）。