《数据结构》课程实验报告

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 专业名称 | 计联培 | 年级 | 大二 | 班级 | 一班 |
| 学生姓名 | 姬彬荃 | 学号 | 201911020125 | 指导教师 | 魏晓超 |
| 实验题目 | 实验1 实现顺序表各种基本运算的算法 | | | 提交时间 | 2020.9.16 |

一、实验目的和要求

（1）熟悉C语言的上机环境，进一步掌握C语言的结构特点。

（2）掌握线性表的顺序存储结构的定义及基本运算

二、实验环境

Visual Studio 2019

三、实验内容及实施

（包括设计思想、源程序、参数说明、采用的关键理论、技术、算法或数据结构等的说明。）

**实验1：顺序表的建立及运算**

实验要求：

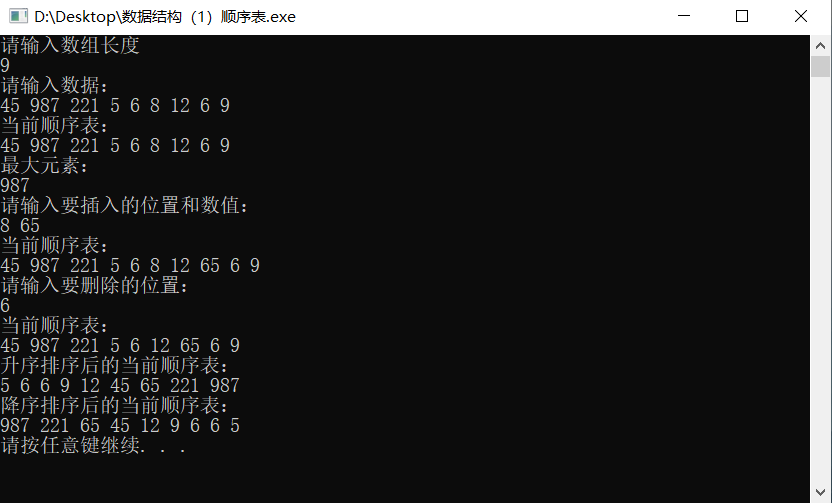
1. 建立一个顺序表，输入n个元素并输出；  
   2、查找线性表中的最大元素并输出；  
   3、在线性表的第i个元素前插入一个正整数x；  
   4、删除线性表中的第j个元素；  
   5、将线性表中的元素按升序排列；  
   \*6、将线性表中的元素就地逆序（只允许用一个暂存单元）；

**【源程序】**

1. /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*
2. \*\* 功能 : 数据结构实验1 实现顺序表各种基本运算的算法
3. \*\* 作者 : 2019Jibinquan
4. \*\* 创建 : 2020-9-16 / 16:02
5. \*\* 联系 : 431263064@qq.com
6. /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/
8. #include <iostream>
9. #include <cstdio>
10. #include <cmath>
11. #include <algorithm>
12. #include <cstring>
13. #include <queue>
14. #include <stack>
15. #include <vector>
16. #include <map>
17. #include <set>
18. #define ios ios::sync\_with\_stdio(false);cin.tie(0);cout.tie(0);
19. #define debug(a) cout << #a << " " << a << endl
20. **using** **namespace** std;
21. **typedef** **long** **long** ll;
22. **const** **double** pi = acos(-1);
23. **const** **double** eps = 1e-8;
24. **const** **int** inf = 0x3f3f3f3f;
25. **const** **int** maxn = 100007;//1e5+7
26. **const** ll mod = 1000000007;//1e9+7
28. **struct** Sqlist {
29. **int** arr[maxn];
30. **int** len;
32. **void** setSqlist(**int** n) {
33. len = n;
34. cout << "请输入数据：" << endl;
35. **for** (**int** i = 1; i <= n; i++) {
36. cin >> arr[i];
37. }
38. }
40. **void** outSqlist() {
41. cout << "当前顺序表：" << endl;
42. **for** (**int** i = 1; i <= len; i++) {
43. cout << arr[i] << " \n"[i == len];
44. }
45. }
47. **int** maxnum() {
48. **int** mx = -inf;
49. **for** (**int** i = 1; i <= len; i++) {
50. mx = max(mx, arr[i]);
51. }
52. **return** mx;
53. }
55. **void** insertnum(**int** pos, **int** x) {
56. **if** (pos > len + 1 || pos < 1) {
57. cout << "输入位置有误" << endl;
58. **return**;
59. }
60. **for** (**int** i = len; i >= pos; i--) {
61. arr[i + 1] = arr[i];
62. }
63. arr[pos] = x;
64. len++;
65. }
67. **void** deletenum(**int** pos) {
68. **if** (pos > len || pos < 1) {
69. cout << "输入位置有误" << endl;
70. **return**;
71. }
72. **for** (**int** i = pos; i < len; i++) {
73. arr[i] = arr[i + 1];
74. }
75. arr[len] = 0;
76. len--;
77. }
79. **void** increasesort() {
80. sort(arr + 1, arr + 1 + len);
81. }
83. **void** dicreasesort() {
84. **int** n = len;
85. **while** (n) {
86. **for** (**int** i = 1; i < n; i++) {
87. **if** (arr[i] < arr[i + 1]) {
88. swap(arr[i], arr[i + 1]);
89. }
90. }
91. n--;
92. }
93. }
95. };
97. /\*----------------------------------\*
98. Main Function
99. \*-----------------------------------\*/
101. **int** main()
102. {
103. **int** n, p, x;
104. Sqlist sl;
105. cout << "请输入数组长度" << endl;
106. cin >> n;
107. sl.setSqlist(n);
108. sl.outSqlist();
109. cout << "最大元素：" << endl;
110. cout << sl.maxnum() << endl;
111. cout << "请输入要插入的位置和数值：" << endl;
112. cin >> p >> x;
113. sl.insertnum(p, x);
114. sl.outSqlist();
115. cout << "请输入要删除的位置：" << endl;
116. cin >> p;
117. sl.deletenum(p);
118. sl.outSqlist();
119. sl.increasesort();
120. cout << "升序排序后的";
121. sl.outSqlist();
122. sl.dicreasesort();
123. cout << "降序排序后的";
124. sl.outSqlist();
125. system("pause");
127. **return** 0;
128. }

**四、实验结果 (程序的执行结果)**

测试程序的运行结果。



**五、实验讨论（可选）**

实验中碰到了哪些问题，如何解决的，有何体会等。