

3.1 顺序栈

顺序栈是操作受限的顺序表。根据栈操作特性，对顺序表定义修改得到顺序栈定义；顺序栈数据元素插入与删除限定栈顶进行，而不能对栈中的第 i 个元素进行操作，这使得顺序栈的基本操作的个数少于顺序表，操作也简单。

一、程序设计简介

本验证程序包括两个文件 `SqStack.h` 和 `SqStack.cpp`。

(1) 头文件 `SqStack.h`，其中包括 2 个内容：

① 顺序栈 (`SqStack`) 的结构定义

② 顺序栈的基本操作实现，包括：初始化顺序栈 `InitStack()` (算法 3.1)、数据元素入栈、销毁顺序栈 `DestroyStack()` (算法 3.2)、数据元素出栈 `Push()` (算法 3.3)、数据元素出栈 `Pop()` (算法 3.4)、取栈顶元素 `GetTop()` (算法 3.5)、清空栈 `ClearStack()`、测栈空 `StackEmpty()`、测栈满 `StackFull()`、显示栈元素 `DispStack()` 等。

(2) 源程序文件 `SqStack.cpp`，通过调用 `SqStack.h` 中定义的操作实现程序的各项功能，并增加了查看栈顶指针功能，以方便观察操作对栈顶的影响及栈空、栈满时的栈顶值。程序功能结构如图 1.3.1 示。

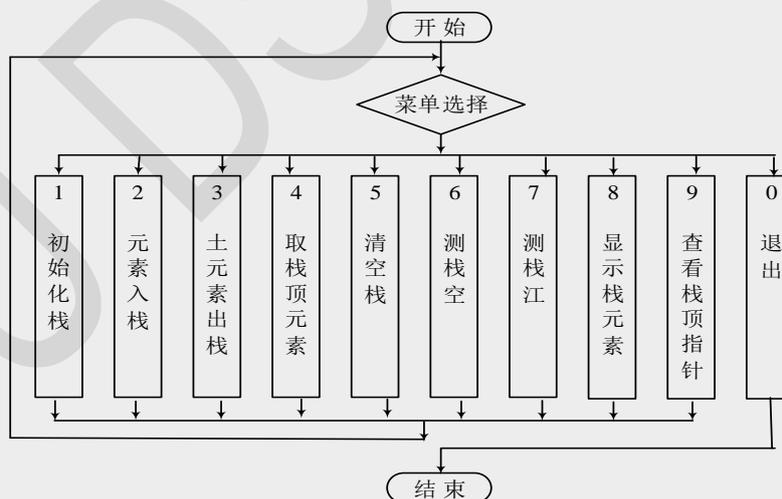


图1.3.1 顺序栈验证实验程序结构框图

二、运行说明

运行程序，显示如图 1.3.2 所示界面。

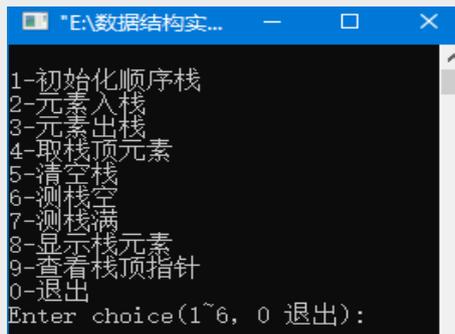


图1.3.2 顺序栈程序运行界面

case 1: 键入“1”，选择功能 1，初始化栈

按屏幕提示，输入要创建的顺序栈容量

case 2: 键入“2”，选择功能 2，元素入栈

2.1 按屏幕提示，输入入栈元素值（1 个整数）

2.2 若栈非满，入栈成功，屏幕显示入栈后栈中所有元素

case 3: 键入“3”，选择功能 3，元素出栈

3.1 屏幕显示出栈元素

3.2 栈非空，出栈成功，屏幕显示出栈元素及出栈后栈中所有元素

case 4: 键入“4”，选择功能 4，取栈顶元素

4.1 栈非空时，屏幕显示栈顶元素

4.2 用功能 8 查看栈中元素；用功能 9 查看出栈前后栈顶的变化。

case 5: 键入“5”，选择功能 5，清空栈

5.1 屏幕显示测试结果

5.2 用功能 9 查看前后栈顶的变化；用功能 7 测栈空；

case 6: 键入“6”，选择功能 6，测栈空

6.1 屏幕显示测试结果

6.2 用功能 8 查看栈中元素，用功能 9 查看栈顶指针，验证的正确性。

case 7: 键入“7”，选择功能 7，测栈满

7.1 屏幕显示测试结果

7.2 用功能 8 查看栈中元素，用功能 9 查看栈顶指针，验证的正确性。

case 8: 键入“8”，选择功能 8，显示栈元素

屏幕显示栈中所有元素

case 9: 键入“9”，选择功能 9，查看栈顶指针

屏幕显示栈顶指针的值

case 0: 键入“0”，选择功能 0，程序退出

屏幕显示“结束运行 bye-bye!”，按任意键，结束程序运行。

三、思考题

1. 研读源程序，回答下列问题

- (1) 验证程序中栈元素类型是什么？如果取其它类型，如何修改程序？
- (2) 验证程序实现了哪些栈的基本操作？对应的函数分别是哪一个？
- (3) 清空栈操作只做了一件事，即 $\text{top}=-1$ ，为什么这样栈就空了？
- (4) 栈空、栈满时栈顶指针分别应该为多少？
- (5) `GetTop()` 操作栈顶指针动了么？
- (6) 栈顶指针为 1 时，栈中有几个元素？

2. 运行程序，回答下列问题

- (7) 元素 11、22、33、44 依次入栈，操作 8 显示栈元素，看到的序列应该是什么？
- (8) 元素入栈、出栈时，栈顶指针如何变化？
- (9) 空栈时，栈顶指针值是什么？
- (10) 对于空栈，执行功能 3、4、6、7、8、9，理解与分析运行结果。
- (11) 对于满栈，执行功能 2、4、7、8、9、5、6，理解与分析运行结果。
- (12) 程序启动成功后直接按“0”退出，会出现什么现象？分析原因。