浙江水利水电学院

实验报告

（18-19学年 第2学期）

|  |  |
| --- | --- |
| 课程名称： | 数据结构与算法 |
| 班 级： | 软工S18-1 |
| 学 号： | 2018b31007 |
| 姓 名： | 曹愉 |
| 实验室（中心）名 称： | 软件实施分室(一) |
| 教学单位： | 信息工程与艺术设计学院 |

2019 年 4 月 1 日

实 验 名 称：栈和队列的应用 指导教师：寿焕君

实 验 日 期：2019.4.1 地 点：现北306

同组学生姓名：无

|  |
| --- |
| 实验内容及要求：  实验内容：   1. 创建顺序栈并进行进栈、出栈操作 2. 创建循环队列并进行进队、出队操作 3. 编写主函数进行测试   实验要求：   1. 熟悉栈和队列的特点及典型应用 2. 掌握栈的顺序存储结构及相应的操作 3. 掌握循环队列的顺序存储结构及相应的操作 4. 会根据具体的需求问题选取合适的数据结构并进行操作 5. 掌握调试程序的方式，培养编程的耐心 |
| 主要仪器名称及型号：   1. 计算机：I5-2400/IntelH61/4G DDR3/500G SATA/HD7350 1GB显存独立显卡 2. 开发工具：Cfree |
| 实验过程：（可附页）  **任务1：库文件的使用**  （1）打开库文件“seqstack.h”，熟悉顺序栈的初始化、判断栈空(top=-1)、进栈和出栈的相关函数。该库文件用于任务2。  （2）打开库文件“seqeq.h”，熟悉顺序栈的建立和输出函数，同时了解循环队列的初始化(front=rear=-1)和输出队列的函数。该库文件用于任务3。  （3）注意：在堆栈中，top指针总是指向栈顶元素的当前位置。在循环队列中，头指针front指向队头元素的前一个位置；尾指针rear总是指向队尾元素。  **任务2：判断括号匹配（保存文件为：学号-1.c）**  判断输入的表达式（即字符串）中包含的圆括号"("和")"、方括号"["和"]"及大括号"{"和"}"是否配对出现，如果配对出现，返回TRUE，否则返回FALSE。  提示：(1)设置一个初始为空的栈，从左至右扫描表达式。  (2)若遇上左括号，则令其入栈；若遇上右括号，则需要与栈顶的左括号进行匹配。  (3)若所遇到的右括号能与栈顶的左括号配对，则令栈顶的左括号出栈，然后继续匹配过程；否则返回0，结束判断过程。  (4)若表达式扫描结束，同时栈变为空，则说明表达式中的括号能匹配，返回1。  **任务3：分离奇偶数（保存文件为：学号-2.c）**  设有顺序栈S，存储了n个整型数据，其中有偶数也有奇数；有一循环队列Q，队列Q的空间大于n ，初始时队列为空。现将顺序栈S中的偶数出栈，并按出栈的先后顺序进入到队列Q中，而奇数仍按原先的先后顺序存储在栈S中。试根据上述要求，编写相应的算法。  提示：奇数可以从front指针位置开始逆时针方向存入循环队列中。  **任务4：成果提交**  实验报告保存文件为：**学号实验报告-栈和队列.docx**，连同2个C源程序文件上传到<ftp://192.168.106.99>上。（不要打包） |
| 实验数据记录或图片：（可附页）   1. 任务2：  |  | | --- | | 关键代码：  for (int i = 0; i < strlen(str); i++)  {  char c = str[i];  if (c == '(' || c == '[' || c == '{') {  push(&s, c);  }  else if (c == ')' || c == ']' || c == '}') {  if (empty(s)) { return 0; }  char v = pop(&s);  if (v == '(' && c == ')') {  continue;  }  else if (v == '[' && c == ']') {  continue; }  else if (v == '{' && c == '}') {  continue; }  else{return 0; }  }  else{  continue; }  }  if (empty(s)){  return 1; }  else{  return 0; } | | 实验结果： |  1. 任务3：  |  | | --- | | 关键代码：  q->front = MAXSIZE;  while (!empty(s)) {  if (s->elem[s->top] % 2 == 0) {  q->rear++;  q->elem[q->rear] = s->elem[s->top];  }  else { //队列末尾倒序插入  q->front--;  q->elem[q->front] = s->elem[s->top];  }  s->top--;  }  while (q->front < MAXSIZE) {  s->top++;  s->elem[s->top] = q->elem[q->front];  q->front++;  }  q->front=-1; | | 实验结果： |  1. 拓展：  |  | | --- | | 用一维数组a[7]顺序存储一个循环队列，队首和队尾指针分别用front和rear表示，当前队列中已有5个元素：22，30，16，36，58，其中22是队首元素，front值为5，请画出对应的存储和首尾指针状态图，当连续做2次出队运算后，再让元素80，55依次进队，请再画出对应的存储和首尾指针状态图。 | | 结果：  初始存储状态 出队入队后存储状态   |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | 0 | 30 |  | 0 |  | <-front | | 1 | 16 |  | 1 | 16 |  | | 2 | 36 |  | 2 | 36 |  | | 3 | 58 | <-rear | 3 | 58 |  | | 4 |  |  | 4 | 80 |  | | 5 |  | <-front | 5 | 55 | <-rear | | 6 | 22 |  | 6 |  |  | | |
| 实验结论：  本实验共有（ 3 ）小题，已完成（ 3 ）小题，完成的题目经运行测试，结果正确。 |
| 教师评语：  该同学在本次实验过程中学习态度认真，具有很强的观察、分析和解决问题的实验能力，能按时完成本次实验的全部任务。经程序运行测试，实验结果正确。  实验完成情况：优；代码规范：优；实验报告表达能力：好。  成绩：优 批阅教师：寿焕君  2019年4月1日 |