**河南开封科技传媒学院实验报告**

••••••••••••••••••••••••••••••••• 密 ••••••••••••••••••••••••••••••••• 封 ••••••••••••••••••••••••••••••••• 线 •••••••••••••••••••••••••••••••••

**20 -20 学年第 学期**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 姓名： | 刘东 | | 学号： | 2136101045 | 专业： | 软件工程 | |
| 所在学院： | | 理工学院 | | | 指导老师： | | 袁帅 |
| 实验时间： | | 2023/3/1 | | | 实验地点： | | 6503 |
| 课程名称： | | 数据结构 | | | | | |
| 实验题目： | | 第二次实验 | | | | | |
| 实验目的： | | | | | | | |
| 实验内容： | | | | | | | |
| 实验代码：  #include<iostream>  using namespace std;  const int N = 1024;  typedef struct SeqList{  int arr[N];  int size;  }SqlList;  //初始化线性表  void InitSqlList(SqlList\* L) {  L->size = 0;  }  //返回线性表的个数  int ListLength(SqlList L) {  return L.size;  }  // 插入元素 1是成功， 0是失败  int insertSqlList(SqlList\* L, int i, int e) {  int k;  if (L->size == N) return 0;  if (i < 1 || i > L->size + 1) return 0;  if (i <= L->size) {  for (int j = L->size - 1; j >= i - 1; j--) {  L->arr[j + 1] = L->arr[j];  }  }  L->arr[i - 1] = e;  L->size++;  return 1;  }  //删除元素 1是成功，0是失败  int delSqlList(SqlList\* L, int i, int \*e) {  int k;  if (L->size == 0) return 0;  if (i < 1 || i > L->size) return 0;  \*e = L->arr[i - 1];  if (i < L->size) {  for (int j = i; j < L->size; j++) {  L->arr[j - 1] = L->arr[j];  }  }  L->size--;  return 1;  }  void printSqlList(SqlList L) {  for (int i = 0; i < L.size; i++) {  cout << L.arr[i] << " ";  }  cout << endl;  }  void test01(SqlList \*L, int a[], int n) {  for (int i = 0; i < n; i++) {  L->arr[L->size] = a[i];  L->size++;  }  printSqlList(\*L);    }  //思考1  void test02(SqlList\* L, int start, int end) {  for (int i = start; i < start + end; i++) {  int e;  delSqlList(L, i, &e);  }  }  //思考2  void test03(SqlList\* L) {  int left = 0, right = L->size;  while (left < right) {  //循环left到偶数  while(L->arr[left] % 2 != 0) {  left++;  }    //循环right到奇数  while(L->arr[right] % 2 == 0) {  right--;  }    //交换  if(left >= right) break;  int \*t;  \*t = L->arr[left];  L->arr[left] = L->arr[right];  L->arr[right] = \*t;  }    }  int main() {  int a[10] = {1,2,3,4,5,6,7,8,9,10};  int i = 5, j = 7, k = 2;    SqlList L;    InitSqlList(&L);    test01(&L, a, sizeof(a)/sizeof(a[0]));    insertSqlList(&L, i, 33);    cout << "（1） 在指定的i = 5位置处插入元素33；" <<endl;  printSqlList(L);    int e;  delSqlList(&L, k, &e);  delSqlList(&L, k - 1, &e);    cout << "（2） 在指定的k = 4位置处删除相应元素和其前面的那个元素；" <<endl;  printSqlList(L);    cout << "此时元素有" << ListLength(L) << "个"<< endl;    for (int p = 0; p < L.size; p++) {  if (L.arr[p] == j) {  cout << "（4） 求出此时给个j = 7值，求出在线性表的哪个位置。" << p << endl;  break;  }  }    cout << endl;  test02(&L, i, k);  cout << "思考1，从线性表中删除自第i = 5个元素开始的k = 2个元素：" <<endl;  printSqlList(L);      test03(&L);  cout << "思考2：" <<endl;  printSqlList(L);    return 0;  } | | | | | | | |
| 实验结果截图及实验心得： | | | | | | | |