|  |  |  |  |
| --- | --- | --- | --- |
| **软件学院实验报告** | | | |
| 姓名： 学号： 专业： 年级： | | | |
| 课程名称 | | 数据结构 | |
| 实验名称 | | 实验1 集合的交、并、差 | |
| 实验的准备阶段 | 实验内容 | （1）实验目的  通过该实验，让学生复习巩固C语言中的循环结构、循环控制条件、分支结构和数组/链表、函数的调用等有关内容，体会到用数组存储集合时，需要记录集合元素的个数，否则输出结果会出现数据越界现象。  （2）实验内容  通过键盘，分别输入两个数据元素类型为正整数的集合A和B，以负数输入为结束条件，输出两个集合的交、并、差。  （3）实验要求  从程序完善性上考虑，集合元素输入时，要有检查元素重复的功能，每个集合中不允许有重复的元素。集合可以用数组也可以用链表存储。  实现交、并、差运算时，分别把代码写成函数的形式，即实现交运算的函数，实现并运算的函数，实现差运算的函数，在主函数中分别调用三个函数。  使用菜单形式对应各个操作，应允许用户反复查看结果，想结束程序时，输入负数结束，使其编成一个完整的小软件。菜单参考示例如下：  1---输入集合A和B  2---求集合A交B  3---求集合A并B  4---求集合A-B  退出，输入一个负数！  （4）验收/测试用例  输入： A={1,2,3,4,5} B={3,4,5,6,7}  要注意输入的过程中，每输入一个元素都要检查输入的这个元素是否和前面的元素重复，如果重复，要求用户重新输入当前元素。  验收测试时要测试这种重复的情况。  输出 A交B={3， 4， 5} A并B={1，2，3，4，5，6，7} A-B={1, 2} | |
| 实验类型 | 验证性 | |
| 实验的重点、难点 | 重点： 数组或线性表的使用  难点： 去重操作和函数调用 | |
| 实验环境 | VC++6.0 | |
| 实验的实施阶段 | 实验步骤及完成任务情况 | 一、设计思想  二、主要源代码  #include <iostream>  using namespace std;  void input(int arr1[],int\*parr1length,int arr2[],int\*parr2length);  void jiao(int arr1[],int arr1length,int arr2[],int arr2length);  void bing(int arr1[],int arr1length,int arr2[],int arr2length);  void cha(int arr1[],int arr1length,int arr2[],int arr2length);  void show(int arr[],int arrlength);  bool exist(int arr[],int arrlength,int number);  int main(){  int flag;  int arr1[100],arr2[100];  int arr1length,arr2length;    cout<<"+------------------------------------+"<<endl;  cout<<"|集合A和B的数据元素类型为整数。 |\n|输入数字1-4进行对应操作， |\n|输入任意其他值退出。 |"<<endl;  cout<<"+------------------------------------+"<<endl;  cout<<"|1---输入集合A和B |"<<endl;  cout<<"|2---求集合A交B |"<<endl;  cout<<"|3---求集合A并B |"<<endl;  cout<<"|4---求集合A-B |"<<endl;  cout<<"+------------------------------------+"<<endl;  cout<<"请输入:";  cin>>flag;  switch(flag){  case 1:  input(arr1,&arr1length,arr2,&arr2length);  break;  case 2:  jiao(arr1,arr1length,arr2,arr2length);  break;  case 3:  bing(arr1,arr1length,arr2,arr2length);  break;  case 4:  cha(arr1,arr1length,arr2,arr2length);  break;  }  while(flag>0&&flag<=4){  cout<<"+------------------------------------+"<<endl;  cout<<"|集合A和B的数据元素类型为整数。 |\n|输入数字1-4进行对应操作， |\n|输入任意其他值退出。 |"<<endl;  cout<<"+------------------------------------+"<<endl;  cout<<"|1---输入集合A和B |"<<endl;  cout<<"|2---求集合A交B |"<<endl;  cout<<"|3---求集合A并B |"<<endl;  cout<<"|4---求集合A-B |"<<endl;  cout<<"+------------------------------------+"<<endl;  cout<<"请输入:";  cin>>flag;  switch(flag){  case 1:  input(arr1,&arr1length,arr2,&arr2length);  break;  case 2:  jiao(arr1,arr1length,arr2,arr2length);  break;  case 3:  bing(arr1,arr1length,arr2,arr2length);  break;  case 4:  cha(arr1,arr1length,arr2,arr2length);  break;  }  }  return 0;  }  void input(int arr1[],int\*parr1length,int arr2[],int\*parr2length){    \*parr1length=0;  \*parr2length=0;  cout<<"请输入整数作为集合的元素，输入非数字值退出"<<endl;  for(int i=0;;i++){  int number=0;  cout<<"正在输入arr1"<<"["<<i<<"]:";  if(cin>>number&&(!exist(arr1,\*parr1length,number))){  arr1[i]=number;  (\*parr1length)++;  }else if(exist(arr1,\*parr1length,number)){  cout<<"集合中已经存在元素 "<<number<<" ,请重新输入"<<endl;  i--;  continue;  }else{  cout<<"输入非整数值,切换到arr2"<<endl;  cin.clear();  cin.sync();  break;  }  }  for(int i=0;;i++){  int number=0;  cout<<"正在输入arr2"<<"["<<i<<"]:";;  if(cin>>number&&(!exist(arr2,\*parr2length,number))){  arr2[i]=number;  (\*parr2length)++;  }else if(exist(arr2,\*parr2length,number)){  cout<<"集合中已经存在元素 "<<number<<" ,请重新输入"<<endl;  i--;  continue;  }else{  cout<<"输入非整数值,退出输入"<<endl;  cin.clear();  cin.sync();  break;  }  }  cout<<"输入完成,下面是结果"<<endl;  cout<<"A:";  show(arr1,\*parr1length);  cout<<"B:";  show(arr2,\*parr2length);  return;  }  void jiao(int arr1[],int arr1length,int arr2[],int arr2length){  int minl=(arr1length<=arr2length)?arr1length:arr2length;  int jiaoji[minl];  int l=0;  cout<<"当前集合各项为:"<<endl;  cout<<"A:";  show(arr1,arr1length);  cout<<"B:";  show(arr2,arr2length);  if(minl==arr1length){  for(int i=0;i<arr1length;i++){  if(exist(arr2,arr2length,arr1[i])){  jiaoji[l]=arr1[i];  l++;  }  }  }else if(minl==arr2length){  for(int i=0;i<arr2length;i++){  if(exist(arr1,arr1length,arr2[i])){  jiaoji[l]=arr2[i];  l++;  }  }  }  cout<<"A交B:";  show(jiaoji,l);  }  void bing(int arr1[],int arr1length,int arr2[],int arr2length){  int bingji[arr1length+arr2length];  int l=0;  cout<<"当前集合各项为:"<<endl;  cout<<"A:";  show(arr1,arr1length);  cout<<"B:";  show(arr2,arr2length);  for(int i=0;i<arr1length;i++){  if(!exist(bingji,l,arr1[i])){  bingji[l]=arr1[i];  l++;  }  }  for(int i=0;i<arr2length;i++){  if(!exist(bingji,l,arr2[i])){  bingji[l]=arr2[i];  l++;  }  }  cout<<"A并B:";  show(bingji,l);  }  void cha(int arr1[],int arr1length,int arr2[],int arr2length){  int chaji[arr1length];  int l=0;  cout<<"当前集合各项为:"<<endl;  cout<<"A:";  show(arr1,arr1length);  cout<<"B:";  show(arr2,arr2length);  for(int i=0;i<arr1length;i++){  if(!exist(arr2,arr2length,arr1[i])){  chaji[l]=arr1[i];  l++;  }  }  cout<<"A-B:";  show(chaji,l);  }  bool exist(int arr[],int arrlength,int number){  for(int i=0;i<arrlength;i++)  {  if(arr[i]==number){  return true;  }  }  return false;  }  void show(int arr[],int arrlength){  cout<<"{ ";  for(int i=0;i<arrlength;i++){  cout<<" "<<arr[i]<<",";  }  cout<<"}"<<endl;  return;  } | |
| 实验结果的处理阶段 | 实验结果总结 |  | |
| 实验结果的运用 |  | |
| 教师评价 | 总评 |  | |
| 得分 |  |