

**实 验 报 告**

**（2023 / 2024 学年 第 一 学期）**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 课程名称 | 离散数学 | | | | | |
| 实验名称 | 编程实现任意集合上二元关系的性质判定 | | | | | |
| 实验时间 | 2023 | 年 | 10 | 月 | 18 | 日 |
| 指导单位 | 计算机学院计算机科学与技术系 | | | | | |
| 指导教师 | 张琳 | | | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 学生姓名 | 张益帆 | 班级学号 | B21150215 |
| 学院(系) | 计算机学院 | 专 业 | 计算机科学与技术 |

**实 验 报 告**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **实验名称** | 编程实现任意集合上二元关系的性质判定 | | | **指导教师** | 张琳 |
| **实验类型** | **验证** | **实验学时** | **4** | **实验时间** | 2023.10.18 |
| 1. **实验目的和要求**   能正确判定任意二元关系的自反性、对称性、传递性、反自反性和反对称性 | | | | | |
| 二、**实验环境(实验设备)**  硬件：微型计算机  软件：Windows 操作系统、Microsoft Visual C++6.0、Java等可视化编程语言 | | | | | |
| **三、实验原理及内容**  说明：这部分内容主要包括：   1. **形式化描述实验中所使用的数据结构和存储结构，给出函数之间的调用关系和数据传递方式；**   input()函数接收输入的矩阵  output()输出关系矩阵  reflexivity()判断自反性  irreflexivity()判断反自反性  symmetry()判断对称性  assymmetry()判断反对称性  transtivity()判断传递性  **2、给出核心算法的C++或Java等语言的源代码，并加上详细注释，分析算法的时间复杂度；**  #include<iostream>  #include<stdlib.h>  #include<cstring>  using namespace std;  int num; //矩阵大小  const int N = 10; //最大矩阵  int matrix[N][N];  //memset(matrix, 0 , sizeof(matrix));  void input(){  cout << "请输入关系矩阵大小 : " << endl;  cin >> num;  cout << "请输入关系矩阵 : " << endl;  for(int i = 0; i < num; i++){  for(int j = 0; j < num; j++){  cin >> matrix[i][j];  }  }  }  void output()  {  cout << "关系矩阵 : " << endl;  for (int i = 0; i < num; i++)  {  for (int j = 0; j < num; j++)  {  cout << matrix[i][j] << " ";  }  cout << endl;  }  cout << endl;  }  void reflexivity()//判断自反性  {  int count = 0;  for (int i = 0; i < num; i++){  if (matrix[i][i] == 1){  count++;  }  }  if (count == num){  cout << "满足自反性" << endl;  }  else{  cout << "不满足自反性" << endl;  }  }  void irreflexivity(){//判断反自反性  int count = 0;  for (int i = 0; i < num; i++){  if (matrix[i][i] == 1){  count++;  }  }  if (count != num){  cout << "满足反自反性" << endl;  }  else{  cout << "不满足反自反性" << endl;  }  }  void symmetry()//判断对称性  {  int count = 0;  for (int i = 0; i < num - 1; i++)  for (int j = 0; j < num - i; j++){  if ((i != j) && (matrix[i][j] == matrix[j][i])){  count++;  }  }  //cout << "cout = " << count << endl;  if (count == num \* (num - 1) / 2){  cout << "满足对称性" << endl;  }  else  {  cout << "不满足对称性"<< endl;  }  }  void assymmetry(){//判断反对称性  int count = 0;  for (int i = 0; i < num - 1; i++)  for (int j = 0; j < num - i; j++){  if ((i != j) && (matrix[i][j] == matrix[j][i])){  count++;  }  }  if (count != num \* (num - 1) / 2){  cout << "满足反对称性" << endl;  }  else  {  cout << "不满足反对称性"<< endl;  }  }  void transtivity()//判断传递性  {  int flag = 1;  for(int i = 0; i < num; i++){  for(int j = 0; j < num; j++){ //遍历矩阵  if(matrix[i][j] == 1){  for(int k = 0; k < num; k++){  if(matrix[j][k] == 1){  if(matrix[i][k] != 1){  flag = 0;  }  }  }  }  }  }  if(flag == 1){  cout << "满足反传递性" << endl;  }  else{  cout << "不满足反传递性" << endl;  }  }  int main(){  input();  output();  reflexivity();  irreflexivity();  symmetry();  assymmetry();  transtivity();  return 0;  } | | | | | |

**实 验 报 告**

|  |
| --- |
| **3、给出测试数据及运行结果、实验相关结论等。**  判断自反性：  根据自反性的定义只需判断对角线元素是否为0，并计数  若计数结果等于关系矩阵大小，则满足自反性，反之则满足反自反性    判断对称性：  遍历数组，i 不等于 j 时判断数组[i][j]和[j][i]是否相等，并计数。因为0时善意认为满足对称性，因此也纳入计数  若计数等于数组上下三角中元素个数则满足对称性，反之不满足    判断传递性  遍历数组，每个元素[i][j]，再遍历[i][k]，若[i][j]，[i][k] == 0，判断[i][k]是否等于0。若遍历之后都满足则满足传递性，反之则不满足 |

**实 验 报 告**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **四、实验小结**（包括问题和解决方法、心得体会、意见与建议等）  中文五号宋体，英文五号Times new roman字体，1.25倍行距  **说明：这部分内容主要包括：在编程、调试或测试过程中遇到的问题及解决方法、本次实验的心得体会、进一步改进的设想等。**  (一)实验中遇到的主要问题及解决方法  判断对称性时，count正确但显示错误  原因：判断条件设为了count == N(N-1)/2，N为矩阵的最大大小，应当改为当前矩阵大小num  （二）实验心得  计算机借助数组模拟矩阵能够更快速精确地判断关系的性质，同时借助关系的性质可以一定程度上简化代码  （三）意见与建议（没有可省略） | | | | | |
| 1. **支撑毕业要求指标点**   支撑毕业要求的指标点为：   * 1-4掌握计算机科学与技术领域的专业知识，能将专业知识用于分析和解决计算机领域复杂工程问题。   √   * 2-1能够应用数学、自然科学和工程科学的基本知识，识别和分析计算机领域复杂工程问题的特征。 | | | | | |
| **六、指导教师评语 (含学生能力达成度的评价)** | | | | | |
| **成 绩** |  | **批阅人** |  | **日 期** |  |

如果不太想写太多字，“指导教师评语”也可以设计为如下的各选择项用打勾形式（仅仅作为一个简单示例，请各课程负责人根据课程和实验情况以及支撑的指标点来自行设定选择项，同一门课程的不同实验评分细则项允许存在不同）：

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **评 分 细 则** | **评分项** | **优秀** | **良好** | **中等** | **合格** | **不合格** |
| **遵守实验室规章制度** |  |  |  |  |  |
| **学习态度** |  |  |  |  |  |
| **算法思想准备情况** |  |  |  |  |  |
| **程序设计能力** |  |  |  |  |  |
| **解决问题能力** |  |  |  |  |  |
| **课题功能实现情况** |  |  |  |  |  |
| **算法设计合理性** |  |  |  |  |  |
| **算法效能评价** |  |  |  |  |  |
| **回答问题准确度** |  |  |  |  |  |
| **报告书写认真程度** |  |  |  |  |  |
| **内容详实程度** |  |  |  |  |  |
| **文字表达熟练程度** |  |  |  |  |  |
| **其它评价意见** |  | | | | |
| **本次实验能力达成评价（总成绩）** |  |  |  |  |  |