

**Lab Report**

**实验报告**

|  |  |
| --- | --- |
| **Course**: | Class Libraries and Data Structures |
| **Semester**: | 1st semester of the academic year **2024-2025** |
| **Major**: | Software Engineering |
| **Class**: | 2023 |
| **Student Name**: |  |
| **Student ID:** |  |
| **Teacher:** | ZHAO, Hengjun (赵恒军) |

**School of Computer and Information Science**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Name | | Recursion and Backtracking Framework  递归和回溯框架 | | | |
| Date | | Oct，2024 | Type | | ☑Confirmatory （验证确认型）  ☑Design（设计型）  🗆Comprehensive（综合型） |
| 1. **Objective & Requirements（实验目的）**    1. Understand the design and implementation principles of recursion and backtracking algorithm to solve a practical problem   理解递归和回溯算法的原理和实现方法   * 1. Understand the object-oriented design of backtracking framework using C++, especially the iterator inner class   理解采用面向对象思想设计通用回溯算法框架的思路，尤其是其中基于迭代器的通用算法的设计思想   * 1. Grasp the use of the backtracking framework to solve a specific problem, i.e. the 8-Queen problem.   掌握面向对象回溯框架的使用，以 8皇后问题为例培养利用回溯框架解决实际问题的能力 | | | | | |
| 1. **Experimental environment (**platform and software**)（实验环境）**   Windows 7 (or higher versions) + Visual Studio 2010 (or higher versions) | | | | | |
| 1. Experimental content and design (Main Content, Procedure, Codes and Results)（此部分应包含每一个实验内容的详细设计，含实验思路、详细实验步骤、核心代码说明等）   Task 1  A chessboard has eight rows and eight columns. In the game of chess, the queen is the most powerful piece: she can attack any piece in her row, any piece in her column, and any piece in either of her diagonals. Develop and validate a program to place eight queens on a chessboard in such a way that no queen is under attack from any other queen.    在8行乘8列的棋盘上，放置8枚皇后棋子，使得任意两个皇后棋子不可互相攻击，即不在同一行、同一列和同一对角线上。如上图所示。  Requirement（要求）:   1. You should use the backtrack framework to solve the 8-Queen problem.   使用所提供通用回溯框架解决该问题。   1. The codes in 4 of the files are fixed and you are not allowed to modify them 2. main.cpp 3. backtrack.h 4. backtrack.cpp 5. application.h   以上4个文件中的代码不允许做任何修改。   1. The codes in 3 of the files are to be implemented： 2. position.h 3. position.cpp 4. application.cpp (including the iterator)   以上3个文件中的代码需要根据实际问题进行实现. | | | | | |
| 1. **Result analysis and discussion**（Analysis of experimental results and summing up the harvest and the existing problems）此部分应包含实验结果，对实验结果的分析，实验收获的总结，实验中存在问题的讨论等；另外，需要回应一下如下思考题：   1. 在Application的内部类Iterator中为什么封装了一个无类型指针，其作用是什么？  2. 如何修改使你的程序以找出这个棋盘上所有可能的八个皇后放置位置？（可以概述基本思路，也可以修改并运行代码得到准确结果。） | | | | | |
| Comments & Evaluation | Content & Design (A-E) | | |  | |
| Procedure & Codes (A-E) | | |  | |
| Results (A-E) | | |  | |
| Analysis & Discussion (A-E) | | |  | |
| Score (A-E):  Feedback comments: | | | | |