

深圳大学实验报告

课程名称 软件工程

项目名称 模块过程设计

学 院 计算机与软件学院

专 业 计算机科学与技术

指导教师 李俊杰

报 告 人 吴嘉楷 学号 2022150168

实验时间 2024 年 10 月 23 日

提交时间 2024 年 10 月 25 日

教务处制

一、实验目的与要求

1. 了解模块过程分析方法;
2. 掌握程序流程图绘制方法;
3. 了解程序流程图绘制工具的使用;

二、实验内容与方法

1. 分析附件中给出的程序源代码
2. 使用 Visio 完成该程序的程序流程图
3. 根据各组的需求分析, 选择一个逻辑功能模块, 选择一个开源源码, 完成主要模块的程序流程图。
4. 本实验需要在实验报告中粘贴绘制的程序流程图, 提交图文并茂的实验报告

三、实验步骤与过程

1. 选定附件中给出的 1000jiechen.cpp 作为被分析的程序源文件
程序源码:

```
#include "stdafx.h"
#define MAX_NUM 1000
#define STORE_SIZE 3000

int main(int argc, char* argv[])
{
    //Initialize the result and Set the last key
    int result[STORE_SIZE] = {1};
    //Begin compute the result
    for(int j = 1; j <= MAX_NUM; j++)
    {
        //Times each key with the new number
        for(i = 0; i < STORE_SIZE; i++)
        {
            result[i] *= j;
        }
        //We will add the
        for(i = 0; i < STORE_SIZE; i++)
        {
            if(result[i] >= 10)
            {
                //Add the result of 10 times to the high key
                result[i + 1] += result[i] / 10;

                //Get the value of current key.
                result[i] = result[i] % 10;
            }
        }
    }
    //print the result
    for(i = STORE_SIZE - 1; i >= 0; i--)
    {
        printf("%d", result[i]);
    }
    //pause
    getchar();
    return 0;
}
```

图 1 cpp 程序源码

程序说明：

这段代码用 C 语言实现了计算 1 到 1000 的阶乘。首先定义了两个宏，MAX_NUM 为 1000，表示计算的上限，STORE_SIZE 为 3000，表示存储结果的数组大小。主函数中，声明并初始化一个 result 数组，用于存储阶乘的每一位，初始化第一个元素的值为 1。

接着，外层循环从 1 遍历到 1000，依次计算每个数字的阶乘。在内层循环中，将当前结果数组的每个元素与当前数字相乘。由于乘法可能会产生进位，因此需要在另一个循环中处理进位逻辑：如果某一位的值大于等于 10，就将其进位加到下一个高位，同时更新当前位为该位的余数。

最后，从高位到低位打印结果，以显示计算得到的阶乘。代码中还包含一个 getchar() 调用，以便在控制台查看结果。由于 C 语言的整数类型的存储长度有限，这种数组方法使得计算大数阶乘成为可能。

2. 使用 draw.io 网页绘制上述 cpp 程序的程序流程图

- (1) draw.io 网页链接：[draw.io \(diagrams.net\)](https://draw.io)
- (2) 新建.drawio 文件，命名为 cpp



图 2 新建 XML 文件

- (3) 绘制程序流程图如下：

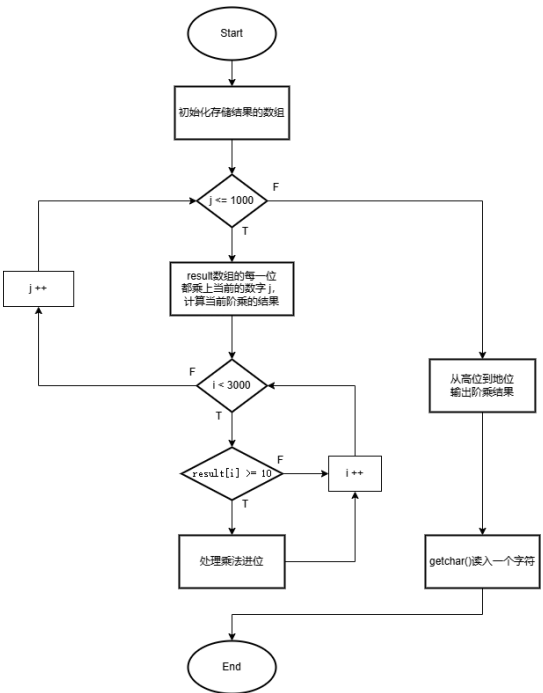


图 3 程序流程图（一）

3. 根据我组功能模块设计程序流程图

(1) 逻辑模块选择：AI 竞赛助手

(2) 模块介绍：

AI 竞赛助手，一款专为竞赛爱好者设计的智能化咨询服务工具，致力于为用户提供全面、精准的竞赛信息咨询与知识解答服务。用户仅需通过自然语言与 AI 竞赛助手进行轻松互动，即可迅速掌握竞赛的核心信息，涵盖竞赛概览、报名通道、参赛资格、日程安排等多个维度。

在平台首页的醒目右下角位置，特设“AI 竞赛助手”悬浮图标作为便捷入口。用户轻触图标，即可瞬间接入与 AI 竞赛助手的实时对话窗口。该助手凭借先进的 AI 大模型技术和云端庞大的竞赛信息数据库，紧密围绕用户的个性化需求，精准生成详尽且贴心的回复，确保每位用户都能获取到高质量的竞赛信息与定制化解决方案。

此外，我们还巧妙融入激励机制，鼓励用户积极利用 AI 助手探索专属竞赛资讯，深度参与平台互动，从而在提升用户体验的同时，有效增强用户的平台粘性，共同营造一个更加活跃、有价值的竞赛交流氛围。

(3) 创建 Alhelper.drawio 的 XML 文件

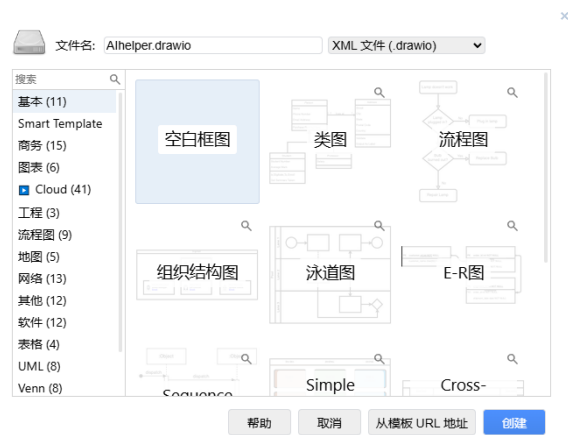


图 4 新建 XML 文件（二）

(4) 新建 Alhelper.drawio 文件

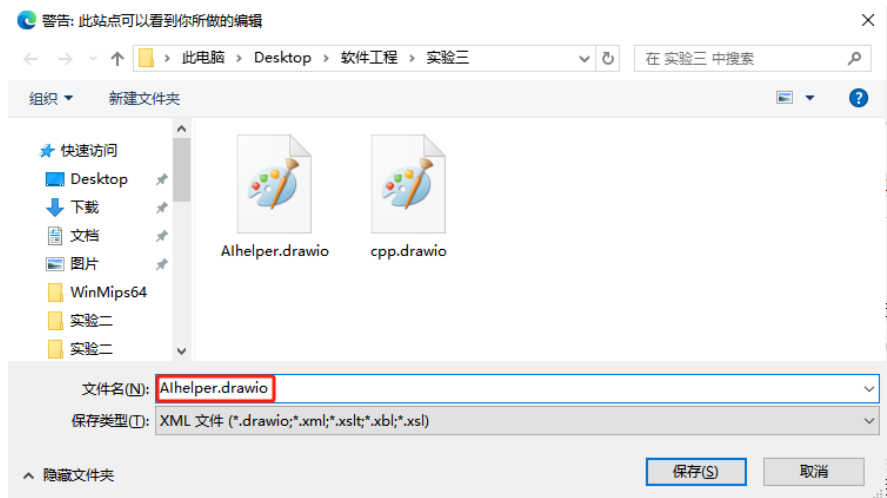


图 5 保存到指定路径

(4) 程序流程设计如下：（使用 draw.io 工具）

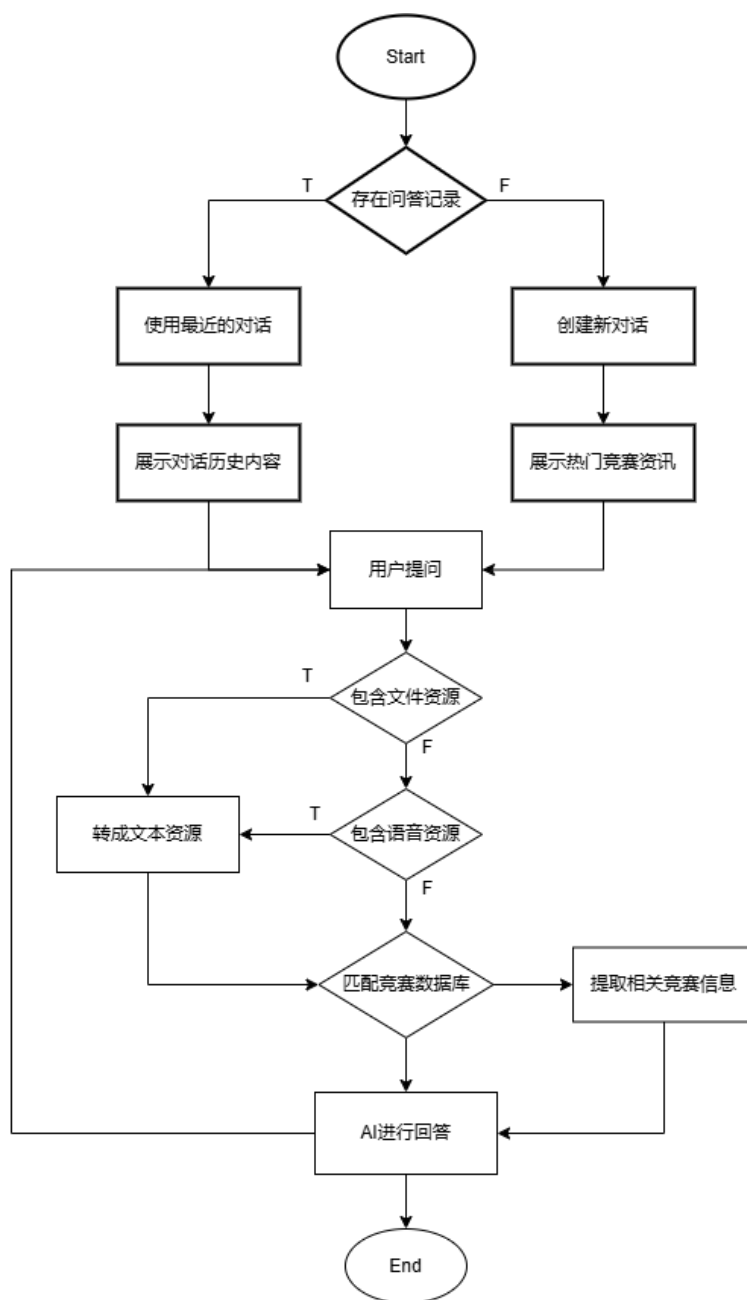


图 6 程序流程图（二）

(5) 程序流程图说明

从起点“Start”开始，系统首先检查是否存在之前的问答记录，以便决定是继续之前的对话还是开启一个新的对话。在这一步骤之后，系统可能会展示之前的对话历史内容，帮助用户回顾之前的讨论，或者展示当前热门的竞赛资讯，以吸引用户的注意力并引导他们提出相关问题。

当用户提出问题时，这些问题可能包含各种文件资源，如图片、文档等，这些资源对于理解和解答问题至关重要。系统会将这些问题和文件资源转换成文本形式，以便进行进一步的处理和分析。如果用户的提问中包含语音内容，系统还会将其转换为文本，以确保所有信

息都能被准确理解和处理。

接下来，系统会利用用户的问题（以及可能的文件或语音资源）来匹配其竞赛数据库。这个数据库包含了大量的竞赛信息，如竞赛名称、规则、日期、地点等。通过匹配数据库，系统能够提取出与用户问题相关的竞赛信息。

最后，AI 会根据提取的竞赛信息生成回答，并呈现给用户。这个回答可能是基于直接匹配的结果，也可能结合了 AI 的推理和判断。一旦回答生成，整个流程就会结束，达到流程图右下角的“End”点。

四、实验结论或体会

实验结论：

在完成了这次软件工程课程的实验后，我对模块过程分析方法有了更深入的理解，同时也掌握了程序流程图的绘制方法。通过这次实验，我不仅学习了如何使用 draw.io 这类工具来绘制程序流程图，还实际应用了这些工具来分析和设计程序。

首先，我学习了如何分析程序源代码，特别是 C 语言编写的 1000jiechen.cpp 程序，它用于计算 1 到 1000 的阶乘。这个程序展示了大数计算的实现方式，通过数组来存储每一位的计算结果，并处理进位逻辑。

在绘制程序流程图的过程中，我使用了 draw.io 工具，它提供了一个直观的界面来帮助我组织和可视化程序的逻辑流程。通过这个工具，我能够将复杂的程序逻辑分解成易于理解 and 管理的步骤。这个过程锻炼了我的逻辑思维能力，也让我更加熟悉了程序流程图的各个组成部分，如决策点、循环、输入输出等。

此外，我还选择了 AI 竞赛助手作为逻辑功能模块，设计了它的程序流程图。通过这个模块，我学习了如何将用户输入转换为系统可以理解的格式，以及如何从数据库中检索信息并生成响应。这个过程不仅提升了我的技术能力，也增强了我的创新思维。

实验难点：

- 理解程序逻辑：**在分析 1000jiechen.cpp 程序时，理解其**大数计算**的逻辑和进位处理机制是一个挑战。我需要仔细阅读代码，理解每一步的计算过程，这对于我的编程能力和逻辑思维是一个考验。比如其中对**乘法进位**的处理操作，就十分巧妙，值得思考。
- 流程图工具的使用：**虽然 draw.io 是强大的工具，但初次使用时，我需要花费一些时间来熟悉它们的功能和操作方式。如何将复杂的程序逻辑以流程图的形式清晰地展示出来，也是一个需要不断练习和改进的过程。
- 模块化设计：**在设计 AI 竞赛助手的程序流程图时，我需要考虑如何将整个系统分解成模块化的组件。这不仅需要技术知识，还需要对系统的整体架构有深入的理解。

（实验报告的篇幅控制在 6-8 页）

指导教师批阅意见：

成绩评定：

指导教师签字：

年 月 日

备注：

注：1、报告内的项目或内容设置，可根据实际情况加以调整和补充。

2、教师批改学生实验报告时间应在学生提交实验报告时间后 10 日内。