计算机科学技术学院实验报告

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **课程名称** | 面向对象程序设计 | | | **学 号** |  |
| **实验项目** | 实验八 | | | **姓 名** |  |
| **学 时** | 2 | **项目性质** | 综合型 | **班 级** | 2405111B |
| **指导教师** | 蒋振刚 | **实验地点** | 实训楼424 | **日 期** | 2025年5月12日 |
| 1. **实验目的和要求**   第8个实验：文件格式解析  目的：练习对已知文件格式文件的读取和写入。  要求：  1）编写一个程序，将一个文本文件拷贝到另外一个文件中。  2）创建一个文本文件，在文件中输出一个九九乘法表。  3）解析一个bmp文件的文件格式信息。并且拷贝到另外一个文件中。将图片中心的9\*9像素区域设置为红绿蓝间隔快。 | | | | | |
| 1. **实验环境**   系统：fedora 42  环境：g++ | | | | | |
| 1. **实验内容与过程**   **#include<iostream>**  **#include<fstream>**  **#include<string>**  **#include<cstdint>**  **using namespace std;**  **// BMP文件头结构体**  **#pragma pack(1)  // 设置1字节对齐**  **struct BMPHeader {**  **char signature[2];     // 文件类型**  **uint32\_t fileSize;     // 文件大小**  **uint16\_t reserved1;    // 保留字段1**  **uint16\_t reserved2;    // 保留字段2**  **uint32\_t dataOffset;   // 数据偏移量**  **};**  **struct BMPInfoHeader {**  **uint32\_t headerSize;   // 信息头大小**  **int32\_t width;        // 图像宽度**  **int32\_t height;       // 图像高度**  **uint16\_t planes;      // 平面数**  **uint16\_t bitsPerPixel;// 每像素位数**  **uint32\_t compression; // 压缩类型**  **uint32\_t imageSize;   // 图像大小**  **int32\_t xPixelsPerMeter;**  **int32\_t yPixelsPerMeter;**  **uint32\_t colorsUsed;**  **uint32\_t colorsImportant;**  **};**  **#pragma pack()**  **void copy(const string& sourceFile, const string& destFile) {**  **fstream file(sourceFile, ios::in | ios::binary);**  **if (!file) {**  **cout << "源文件打开失败！" << endl;**  **return;**  **}**  **fstream copyFile(destFile, ios::out | ios::binary);**  **if (!copyFile) {**  **cout << "目标文件打开失败！" << endl;**  **return;**  **}**  **char ch;**  **while (file.get(ch)) {**  **copyFile.put(ch);**  **}**  **file.close();**  **copyFile.close();**  **}**  **void multiplication() {**  **fstream file;**  **file.open("Multiplication.txt", ios::out);**  **if (!file) {**  **cout << "文件打开失败！" << endl;**  **return;**  **}**  **for (int i = 1; i <= 9; i++) {**  **for (int j = 1; j <= i; j++) {**  **file << i << " \* " << j << " = " << i \* j << "\t";**  **}**  **file << endl;**  **}**  **file.close();**  **}**  **void processBMP(const string& sourceFile, const string& destFile) {**  **fstream inFile(sourceFile, ios::in | ios::binary);**  **if (!inFile) {**  **cout << "BMP文件打开失败！" << endl;**  **return;**  **}**  **// 读取文件头**  **BMPHeader header;**  **BMPInfoHeader infoHeader;**  **inFile.read(reinterpret\_cast<char\*>(&header), sizeof(header));**  **inFile.read(reinterpret\_cast<char\*>(&infoHeader), sizeof(infoHeader));**  **// 显示BMP信息**  **cout << "BMP文件信息：" << endl;**  **cout << "文件大小: " << header.fileSize << " 字节" << endl;**  **cout << "图像宽度: " << infoHeader.width << " 像素" << endl;**  **cout << "图像高度: " << infoHeader.height << " 像素" << endl;**  **cout << "色深: " << infoHeader.bitsPerPixel << " 位" << endl;**  **// 创建输出文件**  **fstream outFile(destFile, ios::out | ios::binary);**  **if (!outFile) {**  **cout << "输出文件创建失败！" << endl;**  **inFile.close();**  **return;**  **}**  **// 复制文件头**  **outFile.write(reinterpret\_cast<char\*>(&header), sizeof(header));**  **outFile.write(reinterpret\_cast<char\*>(&infoHeader), sizeof(infoHeader));**  **// 计算中心位置**  **int centerX = infoHeader.width / 2;**  **int centerY = infoHeader.height / 2;**  **int padding = (4 - (infoHeader.width \* 3) % 4) % 4;**  **// 复制图像数据，修改中心9x9区域**  **for (int y = 0; y < infoHeader.height; y++) {**  **for (int x = 0; x < infoHeader.width; x++) {**  **unsigned char pixel[3];**  **inFile.read(reinterpret\_cast<char\*>(pixel), 3);**  **// 修改中心9x9区域**  **if (x >= centerX - 4 && x <= centerX + 4 &&**  **y >= centerY - 4 && y <= centerY + 4) {**  **int row = (y - (centerY - 4)) % 3;**  **switch (row) {**  **case 0: pixel[0] = 255; pixel[1] = 0; pixel[2] = 0; break;  // 红**  **case 1: pixel[0] = 0; pixel[1] = 255; pixel[2] = 0; break;  // 绿**  **case 2: pixel[0] = 0; pixel[1] = 0; pixel[2] = 255; break;  // 蓝**  **}**  **}**  **outFile.write(reinterpret\_cast<char\*>(pixel), 3);**  **}**  **// 处理填充字节**  **char padding\_data = 0;**  **for (int p = 0; p < padding; p++) {**  **inFile.read(&padding\_data, 1);**  **outFile.write(&padding\_data, 1);**  **}**  **}**  **inFile.close();**  **outFile.close();**  **}**  **int main() {**  **// 测试文件复制**  **copy("work8.txt", "work8\_copy.txt");**    **// 生成乘法表**  **multiplication();**    **// 处理BMP文件**  **processBMP("source.bmp", "modified.bmp");**    **return 0;**  **}** | | | | | |
| 1. **实验结果与分析**   **1. 文件复制功能**  **实现了基本的二进制文件复制功能**  **通过`copy()`函数将"work8.txt"的内容完整复制到"work8\_copy.txt"中**  **使用二进制模式打开文件，确保所有类型的文件都能正确复制**  **采用逐字节复制的方式，保证数据的完整性**  **2. 乘法表生成**  **multiplication()函数成功生成9x9乘法表**  **输出格式整齐，使用制表符(\t)分隔**  **采用递增的方式展示乘法表，每行显示从1到i的乘积**  **结果保存在"Multiplication.txt"文件中**  **3. BMP图像处理**  **成功实现了BMP文件的读取和修改功能**  **关键实现：**  **正确解析BMP文件头和信息头**  **在图像中心位置绘制了9x9的彩色方块**  **处理了行填充（padding）问题，保证图像数据的正确性**  **电脑萤幕画面  AI 生成的内容可能不正确。** | | | | | |
| 1. **实验心得**   通过本次文件操作实验，我深入理解了C++中文件处理的重要概念和技术。特别是在处理BMP图像文件时，学习了二进制文件的读写方法、结构体的字节对齐以及图像数据的处理技巧。在实现乘法表生成和文件复制功能时，掌握了文本文件和二进制文件的不同操作方式。  实验过程中遇到了一些挑战，如BMP文件的padding处理和中心区域的颜色修改，通过查阅资料和反复调试最终解决了这些问题。这让我认识到在实际编程中，细节处理和数据结构的合理设计对程序的正确性至关重要。  这次实验不仅提升了我的编程技能，也加深了对文件系统和图像处理的理解，为今后进行更复杂的文件操作打下了良好基础。 | | | | | |
| 1. **教师评语** | | | | | |
| 1. **实验成绩**   教师签名： 蒋振刚 批阅日期： 2025 年 5 月 12日 | | | | | |

注：项目性质为 演示型、验证型、设计型、综合型和创新型。