

ЛАБОРАТОРНАЯ РАБОТА №6

Курс «Алгоритмизация и программирование»

Тема: Работа с файлами.

Цель: Научиться программно работать с бинарными и текстовыми файлами в файловой системе ОС Windows.

Темы для предварительной проработки ^[УСТНО]:

- Потоки .NET.
- Текстовые файлы в C#.
- Бинарные файлы в C#.

Общие задания ^[КОД] :

1. Дополнить код лабораторной работы №5, организовав хранение данных предметной области в бинарном (четные варианты) или текстовом (нечетные варианты) файле lab.dat. Данные должны считываться из файла при запуске программы и записываться в файл при закрытии программы.
2. Написать программу для работы с бинарными файлами, в соответствии с вариантом (*приложение А*).
3. Написать программу для работы с текстовыми файлами, в соответствии с вариантом (*приложение Б*).
4. Написать программу, которая создает на одном из разделов жесткого диска директорию Lab6_Temp, автоматически копирует в эту директорию Ваш файл lab.dat из задания 1 и создает в ней копию этого файла lab_backup.dat путем побайтового копирования. Вывести на консоль информацию о файле lab.dat: размер, время последнего изменения, время последнего доступа.
5. Написать программу, которая позволяет ввести имя bmp-файла, считать его заголовки и вывести на консоль информацию о размере файла, ширине и высоте в пикселях, количестве бит на пиксель, разрешении горизонтальном и вертикальном (количестве пикселей на метр), типе сжатия (без сжатия / 4бит RLE / 8бит RLE). Подготовьте несколько файлов изображений и проверьте на них Вашу программу. Структура bmp-файла приведена в *приложении В*.

Контрольные вопросы ^[ОТЧЕТ] :

1. Что понимается под потоком в .NET?
2. Что такое файл? Как файлы связаны с потоками?
3. Укажите основные файловые операции.
4. Как организовать чтение/запись для текстовых файлов в .NET?
5. Как организовать чтение/запись для бинарных файлов в .NET?
6. Что означает «прямой доступ» к элементам файла? Как его реализовать?
7. Возможности .NET по работе с файловой системой (создание, удаление, копирование и перемещение файлов и директорий).
8. Для чего служит компонент Path в .NET?

Рекомендуемые источники:

- [1] Шилдт Г. С# 4.0. Полное руководство. – М.: Издательский дом «Вильямс», 2011. – 1056с.
- [2] Албахари Дж., Албахари Б. С# 5.0. Справочник. Полное описание языка. – М.: Издательский дом «Вильямс», 2013. – 1008с.
- [3] Кормен Т., Лейзерсон Ч., Ривест Р. Алгоритмы: Построение и анализ. – М.: МЦНМО, 1999. – 960с.
- [4] Ахо А., Хопкрофт Д., Ульман Д. Структуры данных и алгоритмы. – М.: Издательский дом «Вильямс», 2001. – 384с.
- [5] Кнут Д. Искусство программирования, том 1. Основные алгоритмы. – М.: Издательский дом «Вильямс», 2000. – 720с.

Приложение А. Индивидуальное задание 1.

Вариант 1.

Программно записать в бинарный файл набор пар « $N - 2^N$ » для N от 1 до 100. Написать функцию, которая считывает из этого файла все вторые числа из каждой пары и записывает во второй файл.

Вариант 2.

Программно записать в бинарный файл числа арифметической прогрессии с первым членом 4 и шагом 7. Написать функцию, которая считывает 5-ый и 6-ый члены прогрессии из файла и записывает их во второй файл.

Вариант 3.

Программно записать в бинарный файл числа геометрической прогрессии с первым членом 3 и шагом 4. Написать функцию, которая считывает 3-ый и 7-ый члены прогрессии из файла и записывает их во второй файл.

Вариант 4.

Программно записать в бинарный файл случайно сгенерированную последовательность из 100 чисел. Написать функцию, которая считывает из файла все числа, находит их среднее арифметическое и записывает его во второй файл.

Вариант 5.

Программно записать в бинарный файл набор пар « $N - N^{1/2}$ » для N от 4 до 81. Написать функцию, которая считывает из этого файла все вторые числа из каждой пары и записывает во второй файл.

Вариант 6.

Программно записать в бинарный файл набор пар « $N - \log_2 N$ » для N от 1 до 128. Написать функцию, которая считывает из этого файла все вторые числа из каждой пары и записывает во второй файл.

Вариант 7.

Программно записать в бинарный файл набор троек « M, N, M^N » для M и N от 1 до 5. Написать функцию, которая считывает из этого файла все третьи числа из каждой тройки и записывает во второй файл.

Вариант 8.

Программно записать в бинарный файл набор пар « $N - 1/N$ » для N от 1 до 100. Написать функцию, которая считывает из этого файла все вторые числа из каждой пары и записывает во второй файл.

Вариант 9.

Программно записать в бинарный файл значения функции $f(x) = x * 10 + 27$ для x от 0 до 10 с шагом 0.1. Написать функцию, которая считывает все значения из файла и записывает во второй файл только те значения, которые больше 50.

Вариант 10.

Программно записать в бинарный файл значения функции $f(x) = \sin(\pi x / 180)$ для x от 30 до 90. Написать функцию, которая записывает во второй файл последние 10 чисел из файла.

Приложение Б. Индивидуальное задание 2.

Вариант 1.

Считать текстовый файл, сформировать новый файл, в котором несколько подряд идущих пустых строк заменены одной. Вывести на консоль количество удаленных строк.

Вариант 2.

Считать текстовый файл, сформировать новый файл, в котором оставить только строки, в которых содержится хотя бы одна цифра. Вывести на консоль количество удаленных строк.

Вариант 3.

Считать текстовый файл, сформировать новый файл, в котором заменить все цифры на символ «*». Вывести на консоль количество замен.

Вариант 4.

Считать текстовый файл, сформировать новый файл, в котором за каждой строкой из исходного файла следует строка с количеством символов в этой строке. Вывести на консоль количество строк в исходном файле.

Вариант 5.

Считать текстовый файл, сформировать новый файл, в котором удалить все цифры. Вывести на консоль количество удалений.

Вариант 6.

Считать текстовый файл, сформировать новый файл, в котором заменить все прописные буквы на строчные. Вывести на консоль количество замен.

Вариант 7.

Считать текстовый файл, сформировать новый файл, в котором каждая непустая строка заканчивается копирайтом «(с) Student». Вывести на консоль количество пустых строк в файле.

Вариант 8.

Считать текстовый файл, сформировать новый файл, в котором заменить все символы «<» и «>» на символы «{» и «}». Вывести на консоль количество замен.

Вариант 9.

Считать текстовый файл, сформировать новый файл, в котором попарно слить строки из исходного файла. Вывести на консоль среднюю длину строк в полученном файле.

Вариант 10.

Считать текстовый файл, сформировать новый файл, в котором слить все подряд идущие строки длиной меньше 4 символов в одну строку (остальные строки оставить без изменения). Вывести на консоль количество строк в новом файле.

Приложение В. Структура BMP-файла (заголовки, первые байты файла).

Смещение (+байт)	Длина поля (байт)	Описание поля
Заголовок файла		
0	2	Символы 'BM' (код 4D42h)
2	4	Размер файла в байтах
6	2	0 (Резервное поле)
8	2	0 (Резервное поле)
10	4	Смещение, с которого начинается само изображение (растр).
Заголовок BITMAP (Информация об изображении)		
14	4	Размер заголовка BITMAP (в байтах) равно 40
18	4	Ширина изображения в пикселях
22	4	Высота изображения в пикселях
26	2	Число плоскостей, должно быть 1
28	2	Бит/пиксел 1 = monochrome palette. Кол-во цветов = 2 4 = 4bit palletized. Кол-во цветов = 16 8 = 8bit palletized. Кол-во цветов = 256 16 = 16bit RGB. Кол-во цветов = 65536 24 = 24bit RGB. Кол-во цветов = 16M
30	4	Тип сжатия 0 = BI_RGB (без сжатия) 1 = BI_RLE8 (8 bit RLE сжатие) 2 = BI_RLE4 (4 bit RLE сжатие)
34	4	0 или размер сжатого изображения в байтах.
38	4	Горизонтальное разрешение, пиксел/м
42	4	Вертикальное разрешение, пиксел/м
46	4	Количество используемых цветов
50	4	Количество "важных" цветов.
Палитра (Карта цветов для N цветов), если используется		
54	4*N	Палитра