|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  | | --- | --- | --- | |  |  |  | | МИНОБРНАУКИ РОССИИ | | | | Федеральное государственное бюджетное образовательное учреждение  высшего образования  **«МИРЭА – Российский технологический университет»**  **РТУ МИРЭА** | | | | |
| Институт информационных технологий (ИТ) | |
| Кафедра математического обеспечения и стандартизации информационных технологий (МОСИТ) | |

|  |  |  |  |
| --- | --- | --- | --- |
| **ОТЧЁТ ПО ПРАКТИЧЕСКИМ РАБОТАМ** | | | |
| **по дисциплине «Структуры и алгоритмы обработки данных (часть 2/2)»** | | | |
|  | | | |
| Выполнил студент группы ИКБО-01-22 | | Аблаева А.Т. | |
|  | |  | |
| Принял  *Ассистент* | | Ермаков С.Р. | |
| Практические работы выполнены | «\_\_\_»\_\_\_\_\_\_\_2023 г. | | (подпись студента) | |
| «Зачтено» | «\_\_\_»\_\_\_\_\_\_\_2023 г. | | (подпись преподавателя) | |
|  |  | |  | |

Москва 202

**СОДЕРЖАНИЕ**

[1 ЦЕЛЬ РАБОТЫ 3](#_Toc146324952)

[2 ЗАДАНИЕ 1 3](#_Toc146324953)

[2.1 Формулировка задачи 3](#_Toc146324954)

[2.2 Математическая модель решения 3](#_Toc146324955)

[2.3 Код программы 4](#_Toc146324956)

[2.4 Результаты тестирования 4](#_Toc146324957)

[3 ЗАДАНИЕ 2 5](#_Toc146324958)

[3.1 Формулировка задачи 5](#_Toc146324959)

[3.2 Математическая модель решения 6](#_Toc146324960)

[3.3 Код программы 7](#_Toc146324961)

[3.4 Результаты тестирования 9](#_Toc146324962)

[4 ЗАДАНИЕ 3 10](#_Toc146324963)

[4.1 Формулировка задачи 10](#_Toc146324964)

[4.2 Математическая модель решения 11](#_Toc146324965)

[4.3 Код программы 11](#_Toc146324966)

[4.4 Результат тестирования 12](#_Toc146324967)

[5 ВЫВОД 12](#_Toc146324968)

[6 СПИСОК ИСТОЧНИКОВ 12](#_Toc146324969)

# 1 ЦЕЛЬ РАБОТЫ

Освоить приёмы работы с битовым представлением беззнаковых целых чисел, реализовать эффективный алгоритм внешней сортировки на основе битового массива.

# ЗАДАНИЕ 1

# Формулировка задачи

Задание 1:

**1.а.** Реализуйте вышеприведённый пример, проверьте правильность результата в том числе и на других значениях х.

**1.б.** Реализуйте по аналогии с предыдущим примером установку 7-го бита числа в единицу.

**1.в.** Реализуйте код листинга 1, объясните выводимый программой результат.

# Математическая модель решения

В задании 1.а необходимо установить 5-й бит числа в 0. Алгоритм: побитово перемножаем заданное число с «маской», в которой все разряды, кроме 5, равны 1.

В задании 1.б необходимо установить 7-й бит числа в 1. Алгоритм: побитово складываем заданное число с «маской», в которой все разряды, кроме 7, равны 0.

В задании 1.в представлен код, необходимо объяснить результат его выполнения. Задается целочисленное число x и число n, хранящее количество битов числа x. Также задается число maska, в котором n разрядов и в старшем бит установлен в 1. Выводится битовое представление числа maska. Далее в цикле для каждого из n разрядов получаем побитовое умножение x и maska, со сдвигом до младшего разряда. Выводим получившееся число, представляющее собой бит числа x. В maska сдвигаем 1 на 1 разряд в сторону младшего разряда. В результате работы программы получим битовое представление числа x.

# Код программы

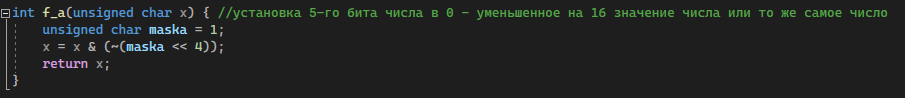


Рис. 1 – код задания 1.а

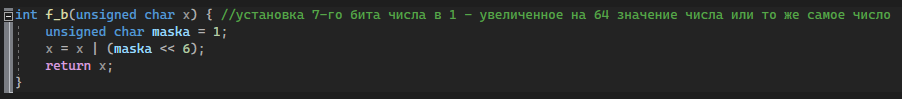


Рис. 2 – код задания 1.б

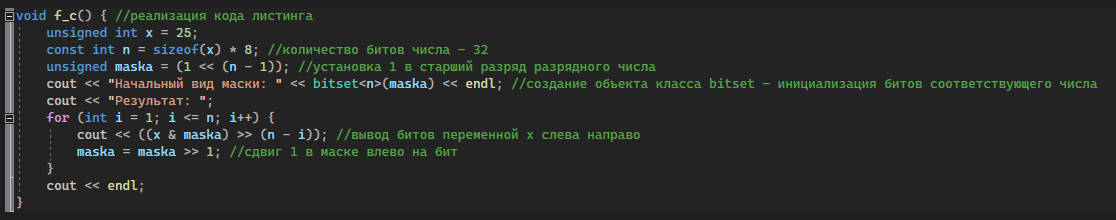


Рис. 3 – код задания 1.в

# Результаты тестирования

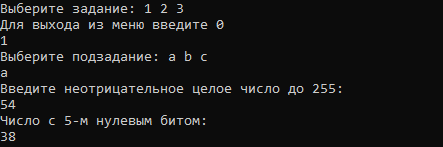


Рис. 4 – результат работы задания 1.а с параметром 54

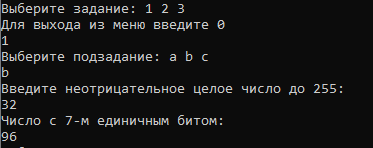


Рис. 5 – результат работы задания 1.б с параметром 32

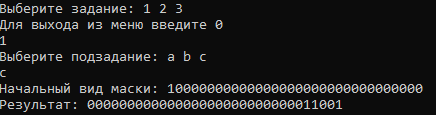


Рис. 6 – результат работы задания 1.в

# ЗАДАНИЕ 2

# Формулировка задачи

Пусть даны не более 8 чисел со значениями от 0 до 7, например, {1, 0, 5, 7, 2, 4}.

Подобный набор чисел удобно отразить в виде 8-разрядной битовой последовательности **11101101**. В ней единичные биты показывают ***наличие*** в исходном наборе числа, равного номеру этого бита в последовательности (нумерация с 0 слева). Т.о. индексы единичных битов в битовом массиве – это и есть числа исходной последовательности.

Последовательное считывание бит этой последовательности и вывод индексов единичных битов позволит естественным образом получить исходный набор чисел ***в отсортированном виде*** – {0, 1, 2, 4, 5, 7}.

Задание 2:

**2.а.** Реализуйте вышеописанный пример с вводом произвольного набора до 8-ми чисел (со значениями от 0 до 7) и его сортировкой битовым массивом в виде числа типа unsigned char. Проверьте работу программы.

**2.б.** Адаптируйте вышеприведённый пример для набора из 64-х чисел (со значениями от 0 до 63) с битовым массивом в виде числа типа unsigned long long.

**2.в.** Исправьте программу задания 2.б, чтобы для сортировки набора из 64-х чисел использовалось не одно число типа unsigned long long, а линейный массив чисел типа unsigned char.

# Математическая модель решения

В задании 2.а необходимо отсортировать массив (максимум 8 чисел со значениями от 0 до 7) используя битовую сортировку. С помощью маски, изначально равной 1, сдвига маски влево на необходимое количество бит и побитового сложения с 0, записываем в соответствующий бит числа типа unsigned char единицу, чтобы записать введенное число. Далее в маске устанавливаем 1 в старший разряд. При прохождении по разрядам числа слева направо и побитовом умножении числа с маской, которая сдвигается вправо на разряд при каждой итерации, выводим число, соответствующее номеру разряда слева направо, начиная с 0. Таким образом получаем отсортированный массив чисел.

В задании 2.б необходимо отсортировать массив (максимум 64 числа со значениями от 0 до 63) используя битовую сортировку. С помощью маски, изначально равной 1, сдвига маски влево на необходимое количество бит и побитового сложения с 0, записываем в соответствующий бит числа типа unsigned long long единицу, чтобы записать введенное число. Далее в маске устанавливаем 1 в старший разряд. При прохождении по разрядам числа слева направо и побитовом умножении числа с маской, которая сдвигается вправо на разряд при каждой итерации, выводим число, соответствующее номеру разряда слева направо, начиная с 0. Таким образом получаем отсортированный массив чисел.

В задании 2.в необходимо отсортировать массив используя битовую сортировку. Составляем массив, состоящий из чисел типа unsigned char. При вводе числа находим индекс элемента массива, в который нужно его записать, с помощью нахождения целой части при делении на 8, и номер бита, который установится в 1, с помощью находения остатка от деления на 8. Проходимся циклом по элементам массива. Во вложенном цикле прохождимся по разрядам числа слева направо и побитово умножаем числа с маской, которая сдвигается вправо на разряд при каждой итерации. Если получаем 1, то выводим число, соответствующее номеру разряда слева направо, начиная с 0, сложенному с произведением 8 и индекса элемента массива. Таким образом получаем отсортированный массив чисел.

# Код программы

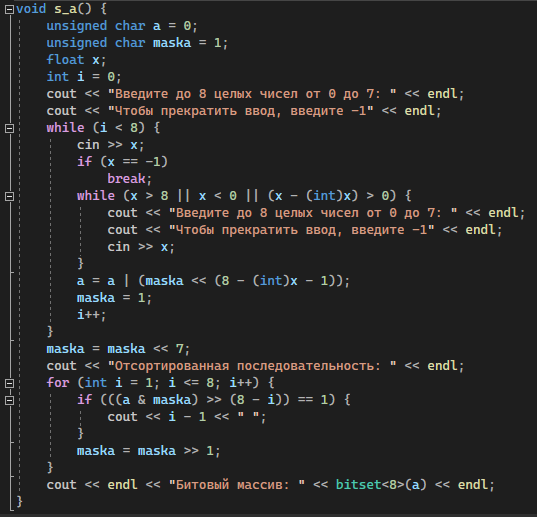


Рис. 7 – код задания 2.а

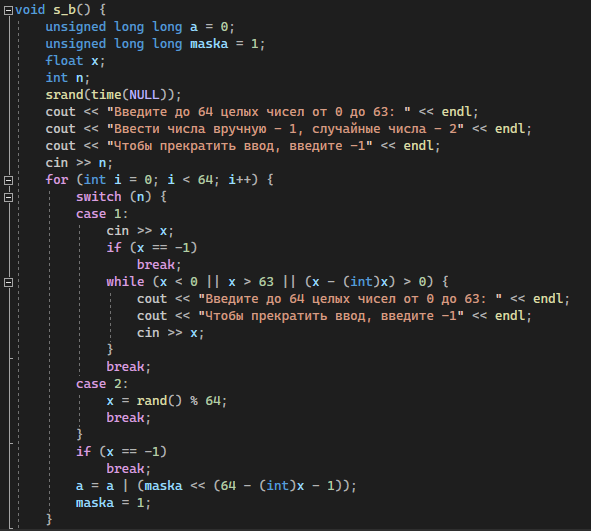


Рис. 8 – код задания 2.б (1/2)

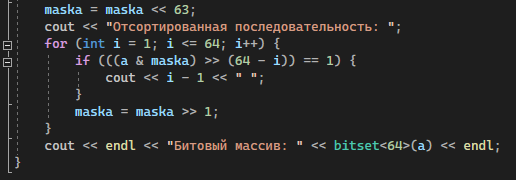


Рис. 9 – код задания 2.б (2/2)

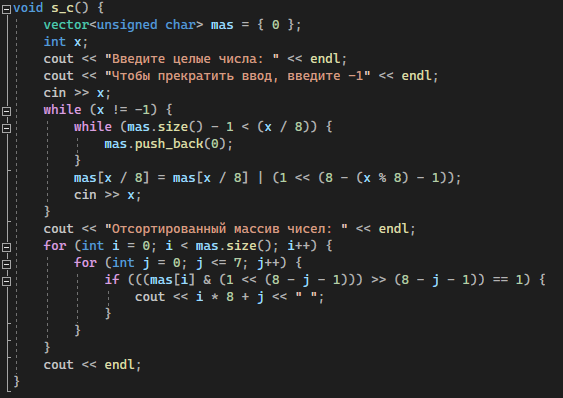


Рис. 10 – код задания 2.в

# Результаты тестирования

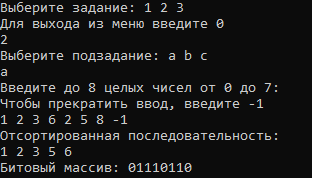


Рис. 11 – результат работы задания 2.а

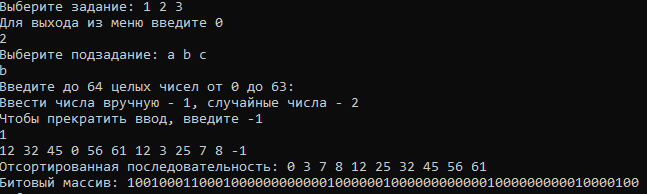


Рис. 12 – результат работы задания 2.б

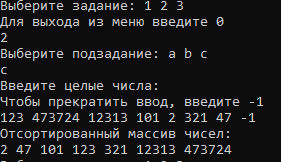


Рис. 13 – результат работы задания 2.в

# ЗАДАНИЕ 3

# Формулировка задачи

Реализовать высокоэффективную сортировку большого объёма числовых данных в файле можно на идее битового массива. Достаточно один раз считать содержимое файла, заполнив при этом в памяти ЭВМ битовый массив и на его основе быстро сформировать содержимое выходного файла в уже отсортированном виде.

При использовании битового массива для представления сортируемых чисел, программу можно представить как последовательность из трех подзадач:

а) Создание битового массива с нулевыми исходными значениями.

б) Считывание целых чисел из файла и установка в 1 соответствующих бит массива.

в) Формирование упорядоченного выходного файла путём последовательной проверки бит массива и вывода в файл номеров (индексов) тех бит, которые установлены в 1.

Задание 3:

**3.а.** Реализуйте задачу сортировки числового файла с заданными условиями. Добавьте в код возможность определения времени работы программы.

**3.б.** Определите программно объём оперативной памяти, занимаемый битовым массивом.

# Математическая модель решения

В задании 3.а необходимо считать из файла 107 чисел, отсортировать их и занести в файл. Составляем массив, состоящий из чисел типа unsigned char, заполненный нулями. При считывании числа из файла находим индекс элемента массива, в который нужно его записать, с помощью нахождения целой части при делении на 8, и номер бита, который установится в 1, с помощью находения остатка от деления на 8. Проходимся циклом по элементам массива. Во вложенном цикле прохождимся по разрядам числа слева направо и побитово умножаем числа с маской, которая сдвигается вправо на разряд при каждой итерации. Если получаем 1, то записываем в файл число, соответствующее номеру разряда слева направо, начиная с 0, сложенному с произведением 8 и индекса элемента массива. Таким образом получаем отсортированный массив чисел.

В задании 3.б необходимо определить программный объём оперативной памяти, занимаемый битовым массивом. Это можно сделать с помощью метода sizeof().

# Код программы

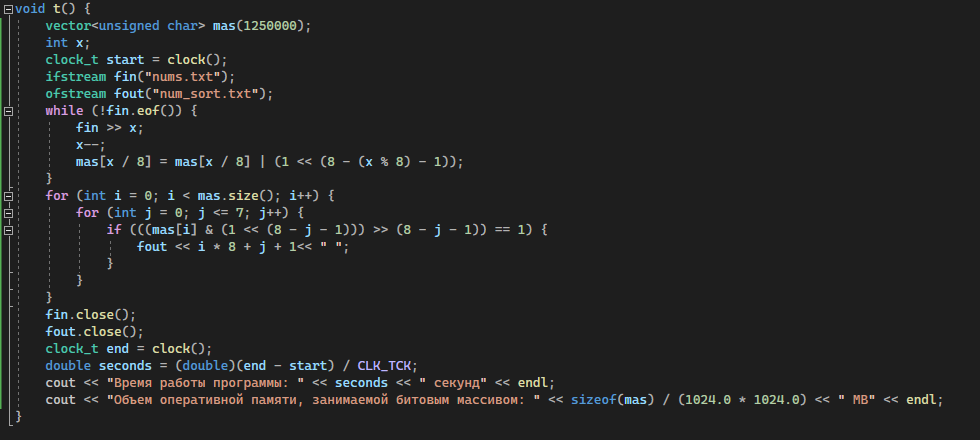


Рис. 14 – код задания 3

# Результат тестирования

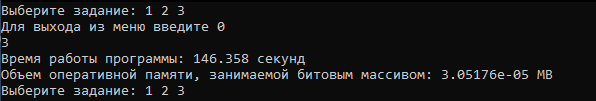


Рис. 15 – результат работы задания 3

# ВЫВОД

В первой практической работе я освоила приёмы работы с битовым представлением беззнаковых целых чисел и реализовала эффективный алгоритм внешней сортировки на основе битового массива.

# 6 СПИСОК ИСТОЧНИКОВ

1. Страуструп Б. Программирование. Принципы и практика с использованием C++. 2-е изд., 2016.
2. Документация по языку С++ [Электронный ресурс]. URL: <https://docs.microsoft.com/ru-ru/cpp/cpp/> (дата обращения 20.09.2023).
3. Курс: Структуры и алгоритмы обработки данных. Часть 2 [Электронный ресурс]. URL: <https://online-edu.mirea.ru/course/view.php?id=4020> (дата обращения 22.09.2023).