|  |
| --- |
| МИНОБРНАУКИ РОССИИ |
| Федеральное государственное бюджетное образовательное учреждениевысшего образования**«МИРЭА − Российский технологический университет»** **РТУ МИРЭА**  |

**Институт информационных технологий (ИИТ)**

**Кафедра практической и прикладной информатики (ППИ)**

**ОТЧЁТ ПО ПРАКТИЧЕСКОЙ РАБОТЕ №1**

**по дисциплине «Структуры и алгоритмы обработки данных»**

|  |  |  |  |
| --- | --- | --- | --- |
| Отчет представлен крассмотрению:Студент группы ИНБО-01-20 | «18» сентября 2021 г. | (подпись) | Бугров Е.П. |
|  |  |  |  |
| Преподаватель | «18» сентября 2021 г. | (подпись) | Сорокин А.В. |

Москва, 2021 г.

Оглавление

[Цель работы. 3](#_Toc82858781)

[Задание. 3](#_Toc82858782)

[Упражнение 1. 3](#_Toc82858783)

[Упражнение 2. 3](#_Toc82858784)

[Упражнение 3. 4](#_Toc82858785)

[Упражнение 4. 4](#_Toc82858786)

[Упражнение 5. 5](#_Toc82858787)

[Код основной функции на Java. 6](#_Toc82858788)

[Тестирование. 7](#_Toc82858789)

[Выводы. 7](#_Toc82858790)

Персональный вариант 14.

# Цель работы.

Получить навыки применения поразрядных операций в алгоритме.

# Задание.

Выполнить упражнения по применению битовых операций по изменению значений битов в ячейке оперативной памяти, созданию маски для изменения значения ячейки. Создание выражения, содержащего поразрядные операции, для выполнения определенной операции над значением ячейки.

## Упражнение 1.

 В первом задании необходимо разработать оператор присваивания и его выражение, которое установит заданные в задании биты исходного значения переменной в значение 1, используя соответствующую маску и поразрядную операцию. Метод для установки в 3-ий, 12-ый и 14-ый справа биты единицу. Для этого мы используем операцию поразрядного «ИЛИ». Маской будет являться число 0х500816 или же 0101 0000 0000 10002, где 1 соответствуют тем битам в которые нам надо установить 1. Реализация метода представлена на рис. 1.



*Рисунок 1 – Метод установки в заданные биты 1*

## Упражнение 2.

 Во втором задании необходимо разработать оператор присваивания и его выражение, которое обнуляет заданные в задании биты исходного значения переменной, используя соответствующую маску и поразрядную операцию. Из варианта следует обнуление всех нечетные битов: 1, 3, 5, 7, 9, 11, 13 и 15-го. Для этого используем операцию поразрядного «И». Маской будет являться число 0xАААА16 или же 1010 1010 1010 10102, где 0 соответствуют позициям, которые нам надо обнулить. Реализация метода представлена на рис. 2.

*Рисунок 2 – Метод обнуления нечетных битов*

## Упражнение 3.

В третьем задании надо разработать оператор присваивания и выражение, которое умножает значение переменной на число, указанное в третьем столбце варианта (512), используя соответствующую поразрядную операцию. Так как 512 это 29, то эту операцию можно реализовать с помощью операции поразрядного сдвига влево на 9. Реализация метода представлена на рис. 3.



*Рисунок 3 – Метод умножения на 512 поразрядным сдвигом*

## Упражнение 4.

В четвертом задании нам надо разработать оператор присваивания и выражение, которое делит значение переменной на число, указанное в четвертом столбце варианта (512), используя соответствующую поразрядную операцию. Так как 512 это 29, то эту операцию можно реализовать с помощью поразрядного сдвига вправо на 9. Для того чтобы исключить переполнение выполним беззнаковый сдвиг вправо с помощью оператора «>>>» Реализация метода представлена на рис. 4.



*Рисунок 4 – Метод деления числа на 512 поразрядным сдвигом*

## Упражнение 5.

 В последнем задании нам необходимо разработать оператор присваивания и выражение, в котором используются только поразрядные операции. В выражении используется маска – переменная. Маска в соответствии с вариантом реализуется единицей в старшем разряде. Необходимо установить n-ый бит в 1. Для этого мы первым шагом выполняем операцию сдвига вправо на 15-n битов для маски. А потом используем операцию поразрядного «ИЛИ» для того, чтобы установить 1. Реализация метода представлена на рис. 5.



*Рисунок 5 – Метод устанавливающий 1 с помощью операций маски и поразрядного сдвига*

## Код основной функции на Java.



*Рисунок 6 – код функции main*

# Тестирование.



# Выводы.

Сегодня мы получили навыки применения поразрядных операций в алгоритме.