|  |  |  |
| --- | --- | --- |
| Министерство науки и высшего образования РФ  Федеральное государственное автономное образовательное учреждение высшего образования  «Пермский государственный национальный исследовательский университет» | | |
|  | Институт компьютерных наук и технологий | |
| **ОТЧЁТ**  по лабораторной работе №1  по дисциплине «Языки программирования»  Вариант 2 | | |
|  | | Работу выполнил  студент группы ДВБ9-ПР2-2023 НБ  2 курса  Малков И.С..  «\_\_\_» \_\_\_\_\_\_\_\_2024 г. |
| Работу проверил  Ракина В. Д.  «\_\_\_» \_\_\_\_\_\_\_\_\_ 2024 г. |
| Пермь 2024 | | |

СОДЕРЖАНИЕ

[Задание 1.1 6](#_Toc178456475)

[Текст задания 6](#_Toc178456476)

[Алгоритм решения 6](#_Toc178456477)

[Тестирование 6](#_Toc178456478)

[Код программы 6](#_Toc178456479)

[Задание 1.3 7](#_Toc178456480)

[Текст задания 7](#_Toc178456481)

[Алгоритм решения 7](#_Toc178456482)

[Тестирование 7](#_Toc178456483)

[Код программы 7](#_Toc178456484)

[Задание 1.5 8](#_Toc178456485)

[Текст задания 8](#_Toc178456486)

[Алгоритм решения 8](#_Toc178456487)

[Тестирование 8](#_Toc178456488)

[Код программы 8](#_Toc178456489)

[Задание 1.7 9](#_Toc178456490)

[Текст задания 9](#_Toc178456491)

[Алгоритм решения 9](#_Toc178456492)

[Тестирование 9](#_Toc178456493)

[Код программы 9](#_Toc178456494)

[Задание 1.9 10](#_Toc178456495)

[Текст задания 10](#_Toc178456496)

[Алгоритм решения 10](#_Toc178456497)

[Тестирование 10](#_Toc178456498)

[Код программы 10](#_Toc178456499)

[Задание 2.1 11](#_Toc178456500)

[Текст задания 11](#_Toc178456501)

[Алгоритм решения 11](#_Toc178456502)

[Тестирование 11](#_Toc178456503)

[Код программы 11](#_Toc178456504)

[Задание 2.3 12](#_Toc178456505)

[Текст задания 12](#_Toc178456506)

[Алгоритм решения 12](#_Toc178456507)

[Тестирование 12](#_Toc178456508)

[Код программы 12](#_Toc178456509)

[Задание 2.5 13](#_Toc178456510)

[Текст задания 13](#_Toc178456511)

[Алгоритм решения 13](#_Toc178456512)

[Тестирование 13](#_Toc178456513)

[Код программы 13](#_Toc178456514)

[Задание 2.7 14](#_Toc178456515)

[Текст задания 14](#_Toc178456516)

[Алгоритм решения 14](#_Toc178456517)

[Тестирование 14](#_Toc178456518)

[Код программы 14](#_Toc178456519)

[Задание 2.9 15](#_Toc178456520)

[Текст задания 15](#_Toc178456521)

[Алгоритм решения 15](#_Toc178456522)

[Тестирование 15](#_Toc178456523)

[Код программы 15](#_Toc178456524)

[Задание 3.1 16](#_Toc178456525)

[Текст задания 16](#_Toc178456526)

[Алгоритм решения 16](#_Toc178456527)

[Тестирование 16](#_Toc178456528)

[Код программы 16](#_Toc178456529)

[Задание 3.3 17](#_Toc178456530)

[Текст задания 17](#_Toc178456531)

[Алгоритм решения 17](#_Toc178456532)

[Тестирование 17](#_Toc178456533)

[Код программы 17](#_Toc178456534)

[Задание 3.5 18](#_Toc178456535)

[Текст задания 18](#_Toc178456536)

[Алгоритм решения 18](#_Toc178456537)

[Тестирование 18](#_Toc178456538)

[Код программы 18](#_Toc178456539)

[Задание 3.7 19](#_Toc178456540)

[Текст задания 19](#_Toc178456541)

[Алгоритм решения 19](#_Toc178456542)

[Тестирование 19](#_Toc178456543)

[Код программы 19](#_Toc178456544)

[Задание 3.9 20](#_Toc178456545)

[Текст задания 20](#_Toc178456546)

[Алгоритм решения 20](#_Toc178456547)

[Тестирование 20](#_Toc178456548)

[Код программы 20](#_Toc178456549)

[Задание 4.1 21](#_Toc178456550)

[Текст задания 21](#_Toc178456551)

[Алгоритм решения 21](#_Toc178456552)

[Тестирование 21](#_Toc178456553)

[Код программы 21](#_Toc178456554)

[Задание 4.3 22](#_Toc178456555)

[Текст задания 22](#_Toc178456556)

[Алгоритм решения 22](#_Toc178456557)

[Тестирование 22](#_Toc178456558)

[Код программы 22](#_Toc178456559)

[Задание 4.5 23](#_Toc178456560)

[Текст задания 23](#_Toc178456561)

[Алгоритм решения 23](#_Toc178456562)

[Тестирование 23](#_Toc178456563)

[Код программы 23](#_Toc178456564)

[Задание 4.7 24](#_Toc178456565)

[Текст задания 24](#_Toc178456566)

[Алгоритм решения 24](#_Toc178456567)

[Тестирование 24](#_Toc178456568)

[Код программы 24](#_Toc178456569)

[Задание 4.9 25](#_Toc178456570)

[Текст задания 25](#_Toc178456571)

[Алгоритм решения 25](#_Toc178456572)

[Тестирование 25](#_Toc178456573)

[Код программы 25](#_Toc178456574)

# Задание 1.1

## Текст задания

Дробная часть.

Дана сигнатура метода: public double fraction (double x); Необходимо реализовать метод таким образом, чтобы он возвращал только дробную часть числа х. Подсказка: вещественное число может быть преобразовано к целому путем отбрасывания дробной части.

Пример:

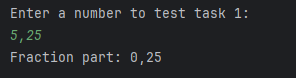
x=5,25

результат: 0,25

# Алгоритм решения

Вычитаем из числа с запятой её целую часть

## Тестирование



## Код программы

https://github.com/IvanMalkS/lab1JavaScript

# Задание 1.3

## Текст задания

Букву в число.

Дана сигнатура метода: public int charToNum (char x); Метод принимает символ х, который представляет собой один из “0 1 2 3 4 5 6 7 8 9”. Необходимо реализовать метод таким образом, чтобы он преобразовывал символ в соответствующее число. Подсказка: код символа ‘0’ — это число 48.

Пример:

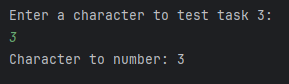
x=’3’

результат: 3

## Алгоритм решения

При помощи конструкции “switch” возвращаем число

## Тестирование



## Код программы

https://github.com/IvanMalkS/lab1JavaScript

# Задание 1.5

## Текст задания

Двузначное.

Дана сигнатура метода: public bool is2Digits (int x); Необходимо реализовать метод таким образом, чтобы он принимал число x и возвращал true, если оно двузначное.

Пример 1:

x=32

результат: true

Пример 2:

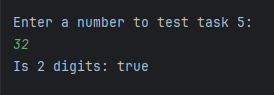
x=516

результат: false

## Алгоритм решения

Проверил что абсолютное значение числа больше 9.и меньше 100

## Тестирование

.

## Код программы

https://github.com/IvanMalkS/lab1JavaScript

# Задание 1.7

## Текст задания

Диапазон.

Дана сигнатура метода: public bool isInRange (int a, int b, int num); Метод принимает левую и правую границу (a и b) некоторого числового диапазона. Необходимо реализовать метод таким образом, чтобы он возвращал true, если num входит в указанный диапазон (включая границы). Обратите внимание, что отношение a и b заранее неизвестно (неясно кто из них больше, а кто меньше)

Пример 1:

a=5 b=1 num=3

результат: true

Пример 2:

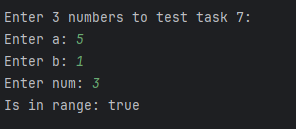
a=2 b=15 num=33

результат: false

## Алгоритм решения

Нашёл максимальное и минимальное из a, b и проверил на вхождения числа в их диапазон.

## Тестирование

.

## Код программы

https://github.com/IvanMalkS/lab1JavaScript

# Задание 1.9

## Текст задания

Равенство.

Дана сигнатура метода: public bool isEqual(int a, int b, int c);

Необходимо реализовать метод таким образом, чтобы он возвращал true, если все три полученных методом числа равны

Пример 1:

a=3 b=3 с=3

результат: true

Пример 2:

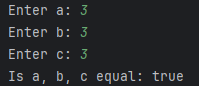
a=2 b=15 с=2

результат: false

## Алгоритм решения

Проверил что первое число равно второму и проверил, что первое число равно третьему.

## Тестирование

.

## Код программы

https://github.com/IvanMalkS/lab1JavaScript

# Задание 2.1

## Текст задания

Модуль числа.

Дана сигнатура метода: public int abs (int x);

Необходимо реализовать метод таким образом, чтобы он возвращал модуль числа х (если оно было положительным, то таким и остается, если он было отрицательным – то необходимо вернуть его без знака минус).

Пример 1:

x=5

результат: 5

Пример 2:

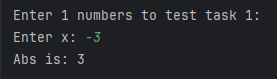
x=-3

результат: 3.

## Алгоритм решения

Через условие вернул обратное значение, если число отрицательное, а если положительно вернул простое число.

## Тестирование

.

## Код программы

https://github.com/IvanMalkS/lab1JavaScript

# Задание 2.3

## Текст задания

Тридцать пять.

Дана сигнатура метода: public bool is35 (int x);

Необходимо реализовать метод таким образом, чтобы он возвращал true, если

число x делится нацело на 3 или 5. При этом, если оно делится и на 3, и на 5, то

вернуть надо false. Подсказка: оператор % позволяет получить остаток от

деления.

Пример 1:

x=5

результат: true

Пример 2:

x=8

результат: false

Пример 3:

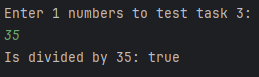
x=15

результат: false

## Алгоритм решения

Проверил делится ли число на 3, проверил делится ли на 5 и исключил вариант, когда число делится и на 3 и на 5 одновременно

## Тестирование

.

## Код программы

https://github.com/IvanMalkS/lab1JavaScript

# Задание 2.5

## Текст задания

Тройной максимум.

Дана сигнатура метода: public int max3 (int x, int y, int z);

Необходимо реализовать метод таким образом, чтобы он возвращал максимальное из трех полученных методом чисел. Подсказка: идеальное решение включает всего две инструкции if и не содержит вложенных if.

Пример 1:

x=5 y=7 z=7

результат: 7

Пример 2:

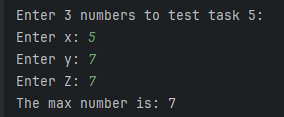
x=8 y=-1 z=4

результат: 8

## Алгоритм решения

Вызвал уже готовый метод “max” для числа y и z и повторно вызвал метод “max” для предыдущего результата и числа x.

## Тестирование

.

## Код программы

https://github.com/IvanMalkS/lab1JavaScript

# Задание 2.7

## Текст задания

Двойная сумма.

Дана сигнатура метода: public int sum2 (int x, int y); Необходимо реализовать метод таким образом, чтобы он возвращал сумму чисел x и y. Однако, если сумма попадает в диапазон от 10 до 19, то надо вернуть число 20.

Пример 1:

x=5 y=7

результат: 20

Пример 2:

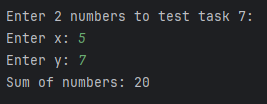
x=8 y=-1

результат: 7

## Алгоритм решения

Сложил два числа и проверил, что результат входит в диапазон от 10 до 19 и при верном результате возвращал 20, а при ложном результате возвращал результат сложения.

## Тестирование

.

## Код программы

https://github.com/IvanMalkS/lab1JavaScript

# Задание 2.9

## Текст задания

День недели.

Дана сигнатура метода: public String day (int x); Метод принимает число x, обозначающее день недели. Необходимо реализовать метод таким образом, чтобы он возвращал строку, которая будет обозначать текущий день недели, где 1- это понедельник, а 7 – воскресенье. Если число не от 1 до 7 то верните текст “это не день недели”. Вместо if в данной задаче используйте switch.

Пример:

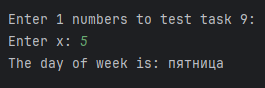
x=5

результат: “пятница”

## Алгоритм решения

Создаем case для каждого дня недели с возвращением строки, которая обозначает текущий день недели.

## Тестирование

.

## Код программы

https://github.com/IvanMalkS/lab1JavaScript

# Задание 3.1

## Текст задания

Числа подряд.

Дана сигнатура метода: public String listNums (int x); Необходимо реализовать метод таким образом, чтобы он возвращал строку, в которой будут записаны все числа от 0 до x (включительно).

Пример:

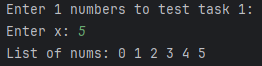
x=5

результат: “0 1 2 3 4 5”

## Алгоритм решения

С помощью цикла for пока счётчик i меньше или равен x добавляем его в строку и добавляем 1 к нему на каждой итерации.

## Тестирование

.

## Код программы

https://github.com/IvanMalkS/lab1JavaScript

# Задание 3.3

## Текст задания

Четные числа.

Дана сигнатура метода: public String chet (int x);

Необходимо реализовать метод таким образом, чтобы он возвращал строку, в которой будут записаны все четные числа от 0 до x (включительно). Подсказа для обеспечения качества кода: инструкцию if использовать не следует.

Пример:

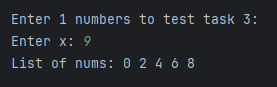
x=9

результат: “0 2 4 6 8”

## Алгоритм решения

С помощью цикла for пока счётчик i меньше или равен x добавляем его в строку и добавляем 2 к нему на каждой итерации.

## Тестирование

.

## Код программы

https://github.com/IvanMalkS/lab1JavaScript

# Задание 3.5

## Текст задания

Длина числа.

Дана сигнатура метода: public int numLen (long x);

Необходимо реализовать метод таким образом, чтобы он возвращал количество знаков в числе x.

Подсказка:

Int у=123/10; // у будет иметь значение 12

Пример:

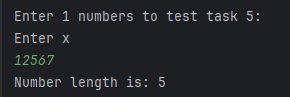
x=12567

результат: 5

## Алгоритм решения

Через цикл while пока остаток от деления не равен нулю, делю число на 10 и считаю количество итераций.

## Тестирование

.

## Код программы

https://github.com/IvanMalkS/lab1JavaScript

# Задание 3.7

## Текст задания

Квадрат.

Дана сигнатура метода: public void square (int x);

Необходимо реализовать метод таким образом, чтобы он выводил на экран квадрат из символов ‘\*’ размером х, у которого х символов в ряд и х символов в высоту.

Пример 1:

x=2

результат:

\*\*

\*\*

Пример 2:

x=4

результат:

\*\*\*\*

\*\*\*\*

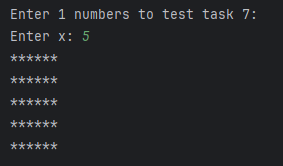
\*\*\*\*

\*\*\*\*

## Алгоритм решения

С помощью двух циклов отрисовываем треугольник, считаем итерацию цикла и выводим количество звезд в строке равное ей.

## Тестирование

.

## Код программы

https://github.com/IvanMalkS/lab1JavaScript

# Задание 3.9

## Текст задания

Правый треугольник.

Дана сигнатура метода: public void rightTriangle (int x);

Необходимо реализовать метод таким образом, чтобы он выводил на экран треугольник из символов ‘\*’ у которого х символов в высоту, а количество символов в ряду совпадает с номером строки, при этом треугольник выровнен по правому краю. Подсказка: перед символами ‘\*’ следует выводить необходимое количество пробелов.

Пример 1:

x=3

результат:

\*

\*\*

\*\*\*

Пример 2:

x=4

результат:

\*

\*\*

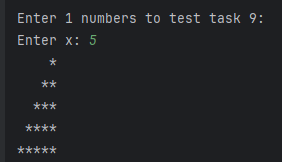
\*\*\*

\*\*\*\*

## Алгоритм решения

С помощью двух циклов отрисовываем треугольник, считаем итерацию цикла и выводим количество звезд в строке равное ей.

## Тестирование

.

## Код программы

https://github.com/IvanMalkS/lab1JavaScript

# Задание 4.1

## Текст задания

Поиск первого значения.

Дана сигнатура метода: public int findFirst (int[] arr, int x); Необходимо реализовать метод таким образом, чтобы он возвращал индекс первого вхождения числа x в массив arr Если число не входит в массив – возвращается -1.

Пример:

arr=[1,2,3,4,2,2,5]

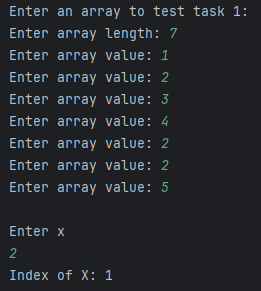
x=2

результат: 1

## Алгоритм решения

Перебираем массив и сравниваем каждое значение по индексу равному итерации цикла. После нахождения сразу возвращаем значения или -1, если цикл завершился.

## Тестирование

.

## Код программы

https://github.com/IvanMalkS/lab1JavaScript

# Задание 4.3

## Текст задания

Поиск максимального.

Дана сигнатура метода: public int maxAbs (int[] arr); Необходимо реализовать метод таким образом, чтобы он возвращал наибольшее по модулю (то есть без учета знака) значение массива arr.

Пример:

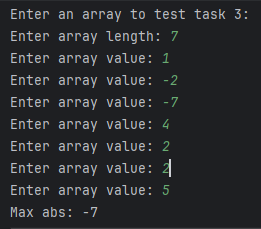
arr=[1,-2,-7,4,2,2,5]

результат: -7

## Алгоритм решения

Объявлем вспомогательные переменные max и result, Перебираем масив циклом for и если абсолютное значение массива больше max присваиваем в max абсолютное значение массива и в result значение массива. После перебора возвращаем result.

## Тестирование

.

## Код программы

https://github.com/IvanMalkS/lab1JavaScript

# Задание 4.5

## Текст задания

Добавление массива в массив.

Дана сигнатура метода: public int[] add (int[] arr, int[] ins, int pos); Необходимо реализовать метод таким образом, чтобы он возвращал новый массив, который будет содержать все элементы массива arr, однако в позицию pos будут вставлены значения массива ins.

Пример:

arr=[1,2,3,4,5]

ins=[7,8,9]

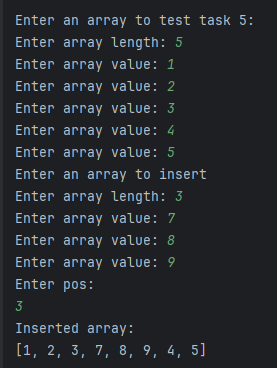
pos=3

результат: [1,2,3,7,8,9,4,5]

## Алгоритм решения

Инициализируем новый массив по длине равный длине arr и ins. Идём циклом for по массиву arr до позиции pos и присваиваем значения arr в новый массив.Дальше идём по массиву ins и присваиваем по позиции (pos + счётчик) для нового массива значения из массива ins. После чего ещё одним циклом for присваиваем оставшиеся значения из массива arr.

## Тестирование

.

## Код программы

https://github.com/IvanMalkS/lab1JavaScript

# Задание 4.7

## Текст задания

Возвратный реверс.

Дана сигнатура метода: public int[] reverseBack (int[] arr); Необходимо реализовать метод таким образом, чтобы он возвращал новый массив, в котором значения массива arr записаны задом наперед.

Пример:

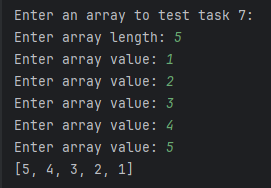
arr=[1,2,3,4,5]

результат: [5,4,3,2,1]

## Алгоритм решения

Создаем вспомогательный массив и циклом for в обратно порядке присваиваем значения из массива arr.

## Тестирование

.

## Код программы

https://github.com/IvanMalkS/lab1JavaScript

# Задание 4.9

## Текст задания

Все вхождения.

Дана сигнатура метода: public int[] findAll (int[] arr, int x); Необходимо реализовать метод таким образом, чтобы он возвращал новый массив, в котором записаны индексы всех вхождений числа x в массив arr.

Пример:

arr=[1,2,3,8,2,2,9]

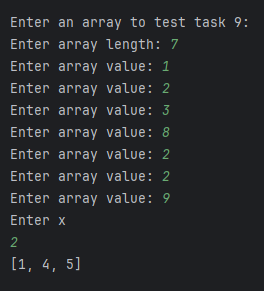
x=2

результат: [1,4,5]

## Алгоритм решения

Создаём вспомогательный массив и счётчик для длины итогового массива. Пробегаемся циклом for по массиву arr и сравниваем его значения с x. В случае совпадения присваиваем индекс элемента в новый массив и добавляем +1 к счётчику, в ином случае сохраняем индекс -1. Создаём новый массив на основе счётчика и циклом for перебираем значения массива с индексами. Если индекс не равен -1 сохраняем значение в результирующий массив.

## Тестирование

.

## Код программы

https://github.com/IvanMalkS/lab1JavaScript