Министерство науки и высшего образования РФ

федеральное государственное бюджетное образовательное

учреждение высшего образования

«Алтайский государственный технический университет им. И.И. Ползунова»

Факультет информационных технологий

Кафедра информационных систем в экономике

Отчёт защищён с оценкой

Преподаватель Убметов. С. В

(подпись) (ФИО)

«\_\_\_\_\_»\_\_\_\_\_\_\_\_\_\_\_2024г.

Отчет по лабораторной работе №3

«Ветвления»

Вариант №7

По дисциплине

«Алгоритмизация и программирование»

ЛР 09.03.03

Студент группы ПИЭ-33 Коршиков И. Д

Преподаватель ассистент, к.т.н Умбетов С. В

Барнаул 2024

## Цель работы:

Изучение разветвляющихся алгоритмов, операторов выбора, программирование разветвляющегося вычислительного процесса.

## Задания:

Реализовать разветвляющийся вычислительный процесс. Самостоятельно решить задачи в соответствии с индивидуальным вариантом.

1. Даны три целых числа: A, B, C. Проверить истинность высказывания: «Число B находится между числами A и C».
2. Даны два числа. Вывести порядковый номер меньшего из них.
3. Вычислить значение функции y=f(x) при произвольных значениях x. Функция y=f(x) показана на рисунке 1.

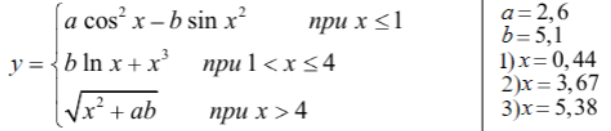


Рисунок 1 – Математическая функция для задания 3

1. Вычислить значение функции y=f(x) при произвольных значениях x. Для вычисления значения функции использовать оператор switch. Функция y=f(x) показана на рисунке 2.

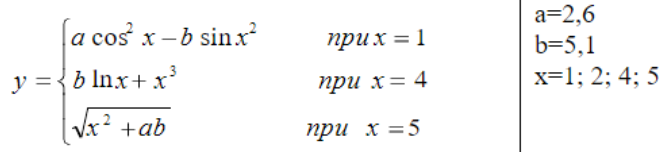


Рисунок 2 – Математическая функция для задания 4

# 

**Задание принял:** Коршиков И.Д

Ход работы

## HTML документы

Для выполнения обоих заданий будет использоваться один шаблон HTML документа с разными заголовками и описанием, он представлен на рисунке 3.



Рисунок 3 – Исходный код HTML

На рисунке 4 представлен скриншот из HTML вадилатора от W3C для него

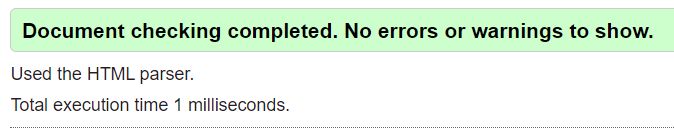


Рисунок 4 – Валидатор HTML

## Задание 1

На рисунке 5 представлена блоксхема алгорима решения к заданию 1

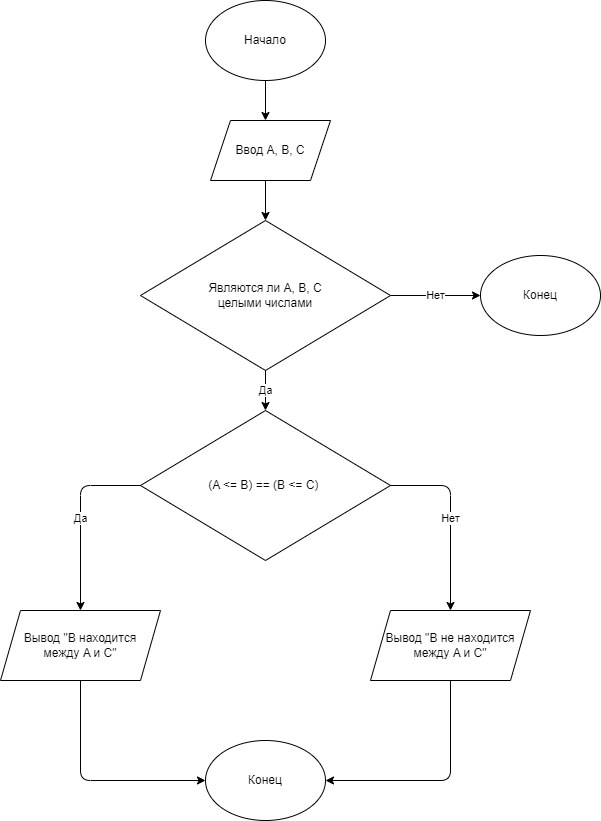


Рисунок 5 – Блок схема задания 1

Исходный код программы представлен на рисунке 6

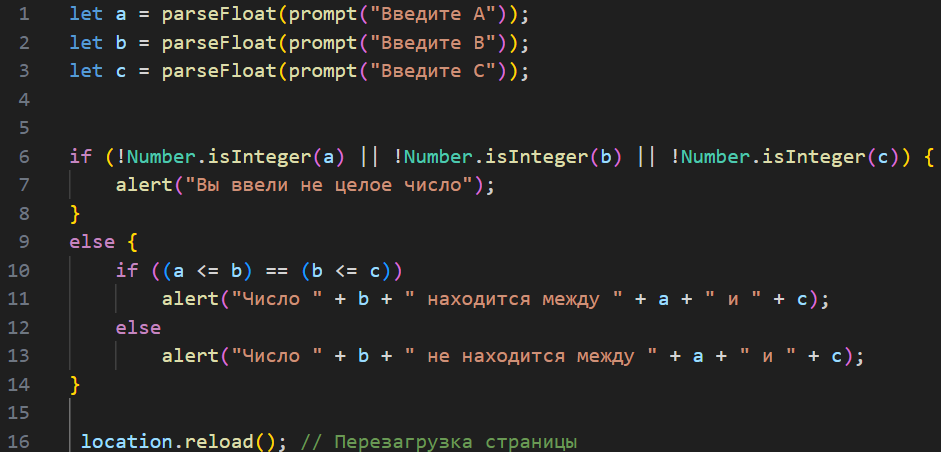


Рисунок 6 – Исходный код программы для задания 2

## Тесты для задания 1

1. На рисунке 7 представлен первый тест



Рисунок 7 – Первый тест для задания 1

1. На рисунке 8 представлен второй тест



Рисунок 8 – Второй тест для задания 1

1. На рисунке 9 представлен третий тест



Рисунок 9 – Третий тест для задания 1

1. На рисунке 10 представлен четвёртый тест



Рисунок 10 – Четвёртый тест для задания 1

1. На рисунке 11 представлен пятый тест



Рисунок 11 – Пятый тест для задания 1

Для проверки результатов тестирования была создана таблица 1

Таблица 1 – Результаты тестирования для задания 1

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Номер теста | Ввод A | Ввод B | Ввод C | Вывод | Ожидаемый вывод |
| 1 | 0 | 1 | 2 | Находится | Находится |
| 2 | 1 | 0 | -3 | Находится | Находится |
| 3 | 1 | 1 | 1 | Находится | Находится |
| 4 | 1 | 100 | 10 | Не находится | Не находится |
| 5 | -10 | -100 | 0 | Не находится | Не находится |

## Задание 2

На рисунке 12 представлена блок схема алгоритма решения к заданию 2

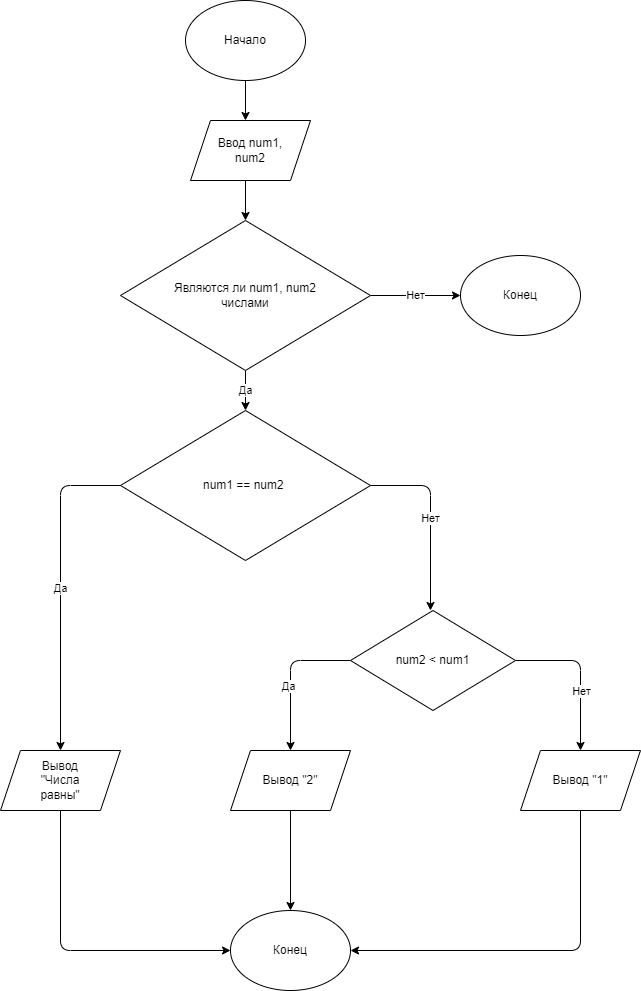


Рисунок 12 – Блок схема к заданию 2

Исходный код программы представлен на рисунке 13

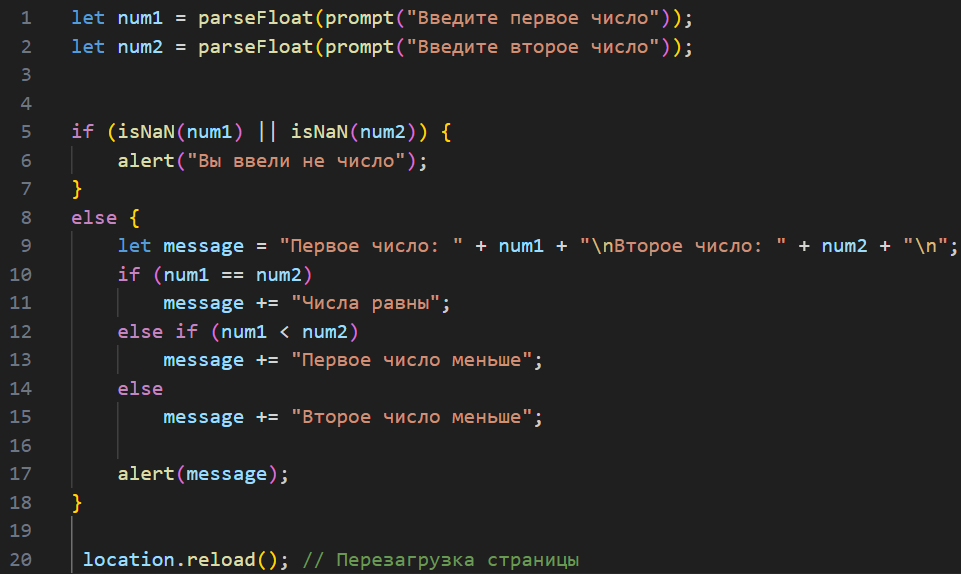


Рисунок 13 – Исходный код программы для задания 2

## Тесты для задания 2

1. На рисунке 14 представлен первый тест

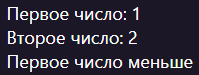


Рисунок 14 – Первый тест для задания 2

1. На рисунке 15 представлен второй тест

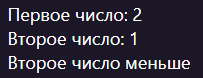


Рисунок 15 – Второй тест для задания 2

1. На рисунке 16 представлен третий тест

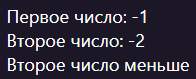


Рисунок 16 – Третий тест для задания 2

1. На рисунке 17 представлен четвёртый тест

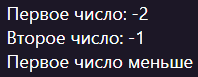


Рисунок 17 – Четвёртый тест для задания 2

1. На рисунке 18 представлен пятый тест

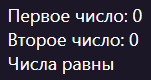


Рисунок 18 – Пятый тест для задания 2

Для проверки результатов тестирования была создана таблица 2

Таблица 2 – Результаты тестирования для задания 2

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Номер теста | Ввод первого числа | Ввод второго числа | Вывод | Ожидаемый вывод |
| 1 | 1 | 2 | Первое | Первое |
| 2 | 2 | 1 | Второе | Второе |
| 3 | -1 | -2 | Второе | Второе |
| 4 | -2 | -1 | Первое | Первое |
| 5 | 0 | 0 | Числа равны | Числа равны |

## Задание 3

На рисунке 19 представлена блок схема алгоритма решения к заданию 3

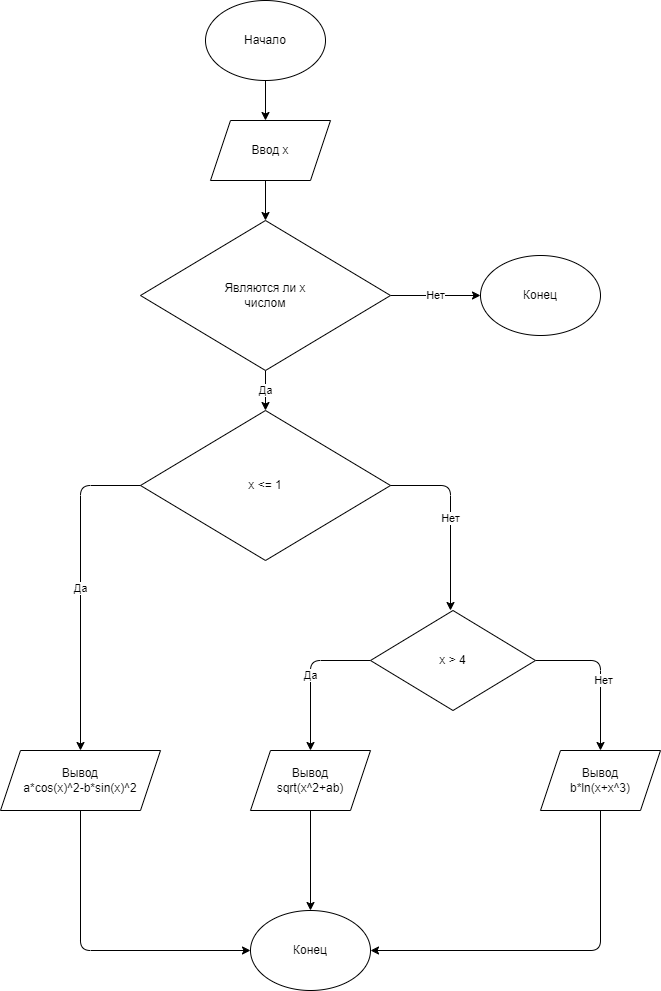


Рисунок 19 – Блок схема к заданию 3

Исходный код программы представлен на рисунке 20

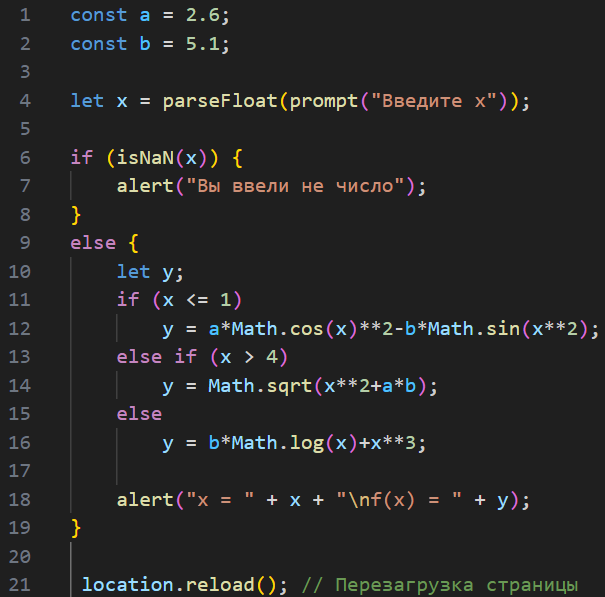


Рисунок 20 – Исходный код программы для задания 3

## Тесты для задания 3

1. На рисунке 21 представлен первый тест



Рисунок 21 – Первый тест для задания 3

1. На рисунке 22 представлен второй тест



Рисунок 22 – Второй тест для задания 3

1. На рисунке 23 представлен третий тест



Рисунок 23 – Третий тест для задания 3

1. На рисунке 24 представлен четвёртый тест



Рисунок 24 – Четвёртый тест для задания 3

1. На рисунке 25 представлен пятый тест



Рисунок 25 – Пятый тест для задания 3

Для проверки результатов тестирования была создана таблица Excel.

Таблица 3 – Результаты тестирования для задания 3

|  |  |  |  |
| --- | --- | --- | --- |
| Номер теста | Ввод x | Вывод y(x) | Ожидаемый вывод y(x) |
| 1 | 0 | 2,6 | 2,6 |
| 2 | 1 | -3,5324 | -3,53249291 |
| 3 | 3 | 32,6029 | 32,60292267 |
| 4 | 4 | 71,0701 | 71,07010124 |
| 5 | 6 | 7,0185 | 7,018546858 |

## Задание 4

На рисунке 26 представлена блок схема алгоритма решения к заданию 4

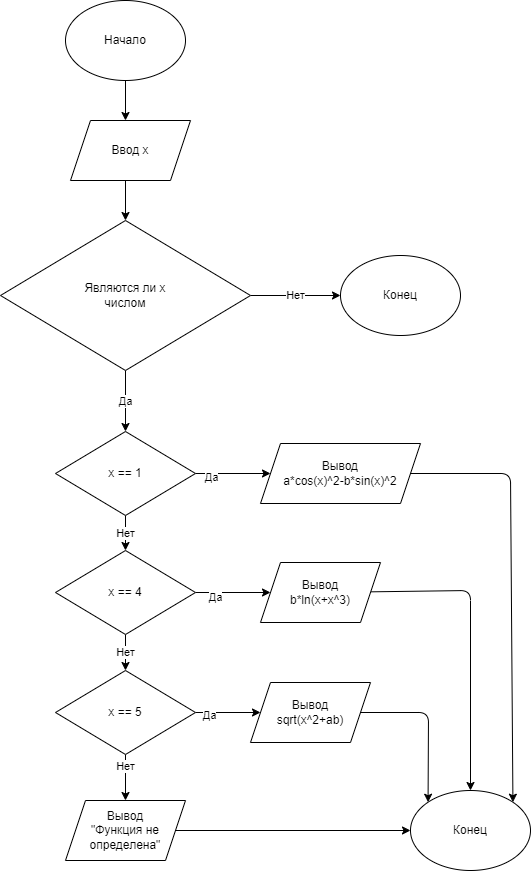


Рисунок 26 – Блок схема к заданию 4

Исходный код программы представлен на рисунке 27

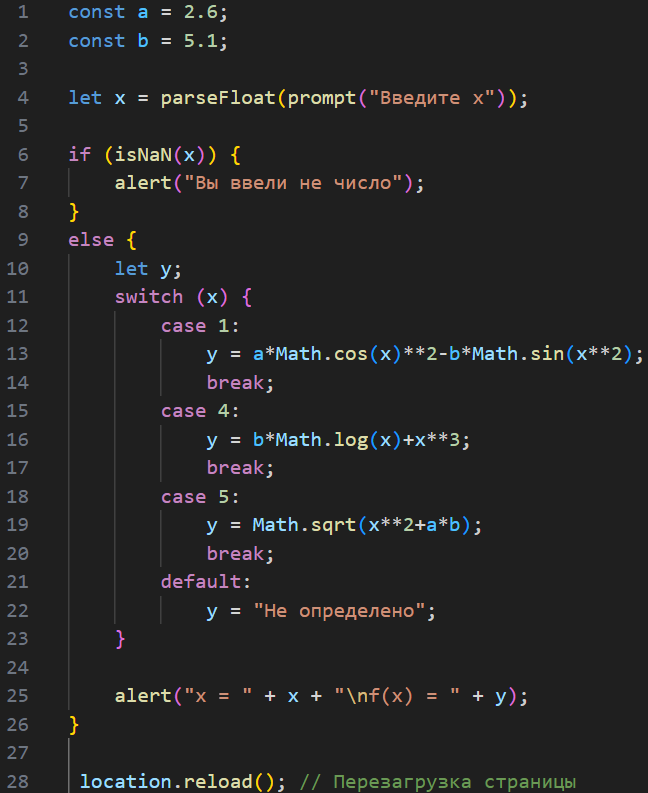


Рисунок 27 – Исходный код программы для задания 4

## Тесты для задания 4

1. На рисунке 28 представлен первый тест



Рисунок 28 – Первый тест для задания 4

1. На рисунке 29 представлен второй тест



Рисунок 29 – Второй тест для задания 4

1. На рисунке 30 представлен третий тест



Рисунок 30 – Третий тест для задания 4

1. На рисунке 31 представлен четвёртый тест



Рисунок 31 – Четвёртый тест для задания 4

1. На рисунке 32 представлен пятый тест



Рисунок 32 – Пятый тест для задания 4

Для проверки результатов тестирования была создана таблица Excel.

Таблица 4 – Результаты тестирования для задания 4

|  |  |  |  |
| --- | --- | --- | --- |
| Номер теста | Ввод x | Вывод y(x) | Ожидаемый вывод y(x) |
| 1 | 1 | -3,53249 | -3,53249291 |
| 2 | 4 | 71,0701 | 71,07010124 |
| 3 | 5 | 6,1854 | 6,185466838 |
| 4 | 0 | Не определено | Не определено |
| 5 | 5,1 | Не определено | Не определено |

# Вывод

В ходе работы я познакомился с операторами ветвления if и switch-case в языке, а также узнал об особенностях операторов сравнения в JavaScript.

Возникли сложности с проверкой преобразования строк в числа. Функции parseInt и parseFloat игнорируют символы, которые идут после цифр, поэтому могут преобразовывать строки вида “213sdfg” в числа (213 в данном случае), что нежелательно в контексте задания. Number() в таких ситуациях возвращает NaN, но может приобразовывать null к 0, что ещё менее приемлемо. Поэтому с целью не перегружать работу множеством проверок я выбрал использовать parseFloat, затем isNaN. С той же целью были опущены проверки на бесконечность isFinite.

## Ссылка на репозиторий:

<https://github.com/k0rshik/korshikov_pie_33_0.git>