|  |  |  |
| --- | --- | --- |
| 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 №5  по дисциплине «Языки программирования»  Вариант 6 | | |
|  | | Работу выполнил  студент группы ПМИ-2/2023 2 курса  Волков А.Н.  «\_\_\_» \_\_\_\_\_\_\_\_2024 г. |
| Работу проверил  Ракина В.Д.  «\_\_\_» \_\_\_\_\_\_\_\_\_ 2024 г. |
| Пермь 2024 | | |

Оглавление

[Задание 1. Время 4](#_Toc181684204)

[Текст задания 4](#_Toc181684205)

[Алгоритм решения 4](#_Toc181684206)

[Тестирование 4](#_Toc181684207)

[Задание 2. Дом 5](#_Toc181684208)

[Текст задания 5](#_Toc181684209)

[Алгоритм решения 5](#_Toc181684210)

[Тестирование 5](#_Toc181684211)

[Задание 3. Сотрудники и отделы 6](#_Toc181684212)

[Текст задания 6](#_Toc181684213)

[Алгоритм 6](#_Toc181684214)

[Тестирование 6](#_Toc181684215)

[Задание 4. Сотрудники и отделы (2) 7](#_Toc181684216)

[Текст задания 7](#_Toc181684217)

[Алгоритм 7](#_Toc181684218)

[Тестирование 7](#_Toc181684219)

[Задание 5. Дом (2) 8](#_Toc181684220)

[Текст задания 8](#_Toc181684221)

[Алгоритм 8](#_Toc181684222)

[Тестирование 8](#_Toc181684223)

[Задание 6. Пистолет 9](#_Toc181684224)

[Текст задания 9](#_Toc181684225)

[Алгоритм 9](#_Toc181684226)

[Тестирование 9](#_Toc181684227)

[Ссылка на GitHub 10](#_Toc181684228)

Задание 1. Шаблоны

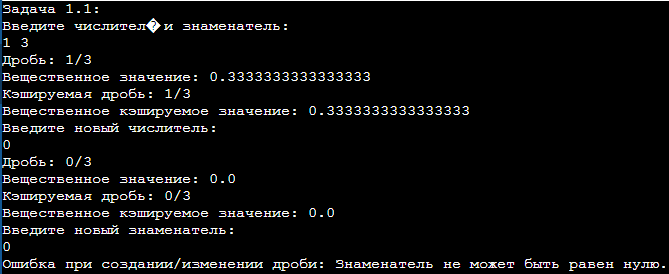
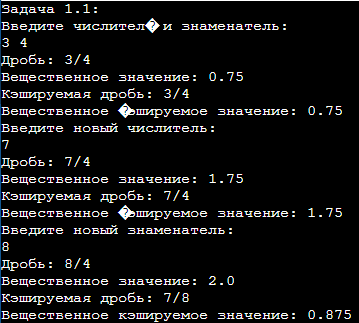
## Текст задания

В класс Дробь, добавить интерфейс на два метода: получение вещественного значения, установка числителя и установка знаменателя. Сгенерировать такую версию дроби, которая будет кэшировать вычисление вещественного значения

## Алгоритм решения

1. Создать класс «Дробь»
2. Создать объект «дробь»
3. Сгенерировать дробь, которая будет кэшировать вычисление вещественного значения
4. Вывести результат

## Тестирование



# Задание 2. Структурные шаблоны

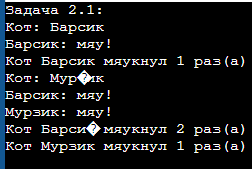
## Текст задания

Необходимо воспользоваться классом Кот и методом принимающим всех мяукающих из задачи 2.5.4. Необходимо таким образом передать кота в указанный метод, что бы после окончания его работы узнать сколько раз мяукал кот за время его работы. На рисунке показан пример работы. Перед вызовом метода создаем кота, отправляем ссылку на кота в метод, после окончания его работы выводим количество мяуканий на экран. Кота изменять нельзя.

## Алгоритм решения

1. Создать класс «Кот»
2. Создать объекты «Кот»
3. Объекты мяукают
4. Вывести результат

## Тестирование



# Задание 3. Список

## Текст задания

Составить программу, которая в списке L из каждой группы подряд идущих одинаковых элементов оставляет только один.

## Алгоритм

1. Создать список
2. Удалить идущие подряд дубликаты
3. Вывести результат

## Тестирование



# Задание 4. Мап

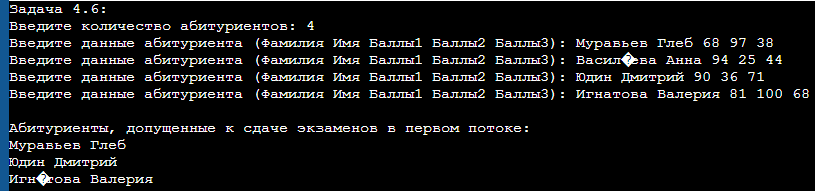
## Текст задания

В некотором вузе абитуриенты проходят предварительное тестирование, по результатам которого могут быть допущены к сдаче вступительных экзаменов в первом потоке. Тестирование проводится по двум предметам, по каждому предмету абитуриент может набрать от 0 до 100 баллов. При этом к сдаче экзаменов в первом потоке допускаются абитуриенты, набравшие по результатам тестирования не менее 30 баллов по каждому из двух предметов. На вход программы подаются сведения о результатах предварительного тестирования. Известно, что общее количество участников тестирования не превосходит 500.

## Алгоритм

1. Создать классы «Applicant» и «TestResultProcessor»
2. Создать объекты «Applicant»
3. Вывести результат

## Тестирование



# Задание 5. Сет

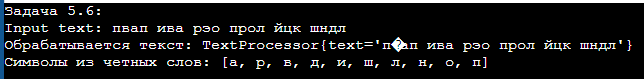
## Текст задания

Файл содержит текст на русском языке. Напечатать в символы, которые встречаются хотя бы однажды в словах с чётными номерами (нумерацию вести с 1).

## Алгоритм

1. Прочитать строку
2. Вывести результат

## Тестирование



# Задание 6. Очередь

## Текст задания

Напечатать в обратном порядке элементы непустой очереди L.

## Алгоритм

1. Создать очередь
2. Вывести результат

## Тестирование



# Задание 7.1. Стрим

## Текст задания

Необходимо написать стрим: Дан набор объектов типа Point, необходимо взять все Point в разных координатах, убрать с одинаковыми X,Y, отсортировать по X, отрицательные Y сделать положительными и собрать это все в ломаную (объект типа Polyline).

## Алгоритм

1. Создать точки
2. Удалить дубликаты
3. Отсортировать по X
4. Взять все значения Y по модулю
5. Вывести результат

## Тестирование



# Задание 7.2. Стрим

## Текст задания

Дан текстовый файл со строками, содержащими имя человека и его номер в следующей форме: Вася:5 Петя:3 Аня:5 Номера людей могут повторяться. У каких-то людей может не быть номера. Необходимо написать стрим выполняющую следующее: читаются все люди из файла, все имена приводится к нижнему регистру, но с первой буквой в верхнем регистре, убираем из перечня всех людей без номеров, а имена оставшихся группируются по их номеру.

## Алгоритм

1. Прочитать файл
2. Убрать людей без номеров
3. Сгруппировать по номерам
4. Вывести результат

## Тестирование



# Ссылка на GitHub

<https://github.com/kuv4lda/PSU_Java/blob/main/lab5/>