|  |  |  |
| --- | --- | --- |
| 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 №4  по дисциплине «Языки программирования»  Вариант 8 | | |
|  | | Работу выполнил  студент группы ПМИ-1,2-2023 2 курса  Хисматов Е.С.  «03» декабря 2024 г. |
| Работу проверил  Ракина В. Д.  «\_\_\_» \_\_\_\_\_\_\_\_\_ 2024 г. |
| Пермь 2024 | | |

СОДЕРЖАНИЕ

[Задание 1 4](#_Toc181709124)

[Текст задания 4](#_Toc181709125)

[Алгоритм решения 4](#_Toc181709126)

[Тестирование 4](#_Toc181709127)

[Код программы 4](#_Toc181709128)

[Задание 2 5](#_Toc181709129)

[Текст задания 5](#_Toc181709130)

[Алгоритм решения 5](#_Toc181709131)

[Тестирование 5](#_Toc181709132)

[Код программы 5](#_Toc181709133)

[Задание 3 6](#_Toc181709134)

[Текст задания 6](#_Toc181709135)

[Алгоритм решения 7](#_Toc181709136)

[Тестирование 7](#_Toc181709137)

[Код программы 7](#_Toc181709138)

[Задание 4 8](#_Toc181709139)

[Текст задания 8](#_Toc181709140)

[Алгоритм решения 8](#_Toc181709141)

[Тестирование 8](#_Toc181709142)

[Код программы 8](#_Toc181709143)

[Задание 5 9](#_Toc181709144)

[Текст задания 9](#_Toc181709145)

[Алгоритм решения 9](#_Toc181709146)

[Тестирование 9](#_Toc181709147)

[Код программы 9](#_Toc181709148)

[Задание 6 10](#_Toc181709149)

[Текст задания 10](#_Toc181709150)

[Алгоритм решения 10](#_Toc181709151)

[Тестирование 10](#_Toc181709152)

[Код программы 10](#_Toc181709153)

[Задание 7 11](#_Toc181709154)

[Текст задания 11](#_Toc181709155)

[Алгоритм решения 11](#_Toc181709156)

[Тестирование 11](#_Toc181709157)

[Код программы 11](#_Toc181709158)

[Задание 8 12](#_Toc181709159)

[Текст задания 12](#_Toc181709160)

[Тестирование 12](#_Toc181709161)

[Код программы 12](#_Toc181709162)

[Задание 9 13](#_Toc181709163)

[Текст задания 13](#_Toc181709164)

[Тестирование 13](#_Toc181709165)

[Код программы 13](#_Toc181709166)

[Задание 10 14](#_Toc181709167)

[Текст задания 14](#_Toc181709168)

[Алгоритм решения 14](#_Toc181709169)

[Тестирование 14](#_Toc181709170)

[Код программы 14](#_Toc181709171)

[Задание 11 15](#_Toc181709172)

[Текст задания 15](#_Toc181709173)

[Алгоритм решения 15](#_Toc181709174)

[Тестирование 15](#_Toc181709175)

[Код программы 15](#_Toc181709176)

# Задание 1

## Текст задания

Обобщенная коробка.

Создайте сущность Коробка, которая обладает следующими характеристиками:

• Может хранить один произвольный объект в один момент времени.

• Объект можно получить и разместить на хранение в любой момент времени.

• Если объект забирают из коробки – ссылку на этот объект необходимо обнулить.

• Если объект кладут в коробку, но она не пуста – необходимо выкинуть исключение.

• Имеет метод проверки на заполненность.

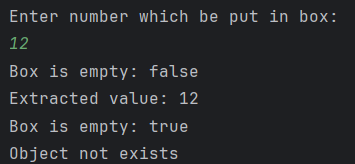
• Методы класса должны работать с тем типом данных, который был указан во время создания объекта

Создайте Коробку которая может хранить целочисленное значение, разместите туда число 3. Передайте Коробку в какой-либо метод, извлеките значение, и выведите его на экран.

## Алгоритм решения

Создать класс Box, параметризированный типом T. Этот тип будет хранить объект. Реализовать метод put, который добавляет объект в коробку, если она пуста, и выбрасывает исключение, если уже есть объект. Создать метод get, который извлекает объект из коробки, обнуляя ссылку на него, и выбрасывает исключение, если коробка пуста. Реализовать метод isEmpty, который проверяет, пуста ли коробка.

## Тестирование



## Код программы

[https://github.com/NTHrdd/Lab4](https://github.com/NTHrdd/Lab3)

# Задание 2

## Текст задания

Сравнимое.

Создайте ссылочный тип Сравнимое, гарантирующий наличие по данной ссылке метода со следующими характеристиками:

• Называется “сравнить”

• Принимает объект.

• Тип принимаемого объекта может быть изменен без изменения самого Сравнимого.

• Возвращает целое число.

## Алгоритм решения

Создать интерфейс Comparable<T>, в котором будет метод compareTo, принимающий объект типа T и возвращающий целое число. Этот интерфейс может быть реализован различными классами, и тип T может быть изменен. Создать класс Person, который реализует интерфейс Comparable<Person>, определяя логику сравнения двух объектов этого типа (например, по возрасту). Метод compareTo должен возвращать целое число.

## Тестирование









## Код программы

[https://github.com/NTHrdd/Lab4](https://github.com/NTHrdd/Lab3)

# Задание 3

## Текст задания

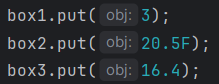
Поиск максимума.

Создайте метод, принимающий набор Коробок из задачи 3.1.1 и возвращающий максимальное из их значений в формате double. Принимаемые методом Коробки могут быть параметризованы любыми видами чисел

## Алгоритм решения

Создать метод findMaxValue, который принимает список коробок, каждая из которых содержит объект, расширяющий Number (например, Integer, Double, Float и т.д.). Для каждой коробки извлекается значение с помощью метода get и обновляйте максимальное значение. Вернуть максимальное значение.

## Тестирование





## Код программы

[https://github.com/NTHrdd/Lab4](https://github.com/NTHrdd/Lab3)

# Задание 4

## Текст задания

Функция.

Разработайте такой метод, который будет принимать список значений типа T, и объект имеющий единственный метод apply. Данный метод надо применить к каждому элементу списка, и вернуть новый список значений типа P, при этом типы T и P могут совпадать, а могут не совпадать. Используйте разработанный метод следующим образом:

1. Передайте в метод список со значениями:“qwerty”, “asdfg”, “zx”, а получите список чисел, где каждое число соответствует длине каждой строки.

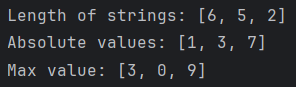
2. Передайте в метод список со значениями: 1,-3,7, а получите список в котором все отрицательные числа стали положительными, а положительные остались без изменений

3. Передайте в метод список состоящий из массивов целых чисел, а получите список в котором будут только максимальные значения каждого из исходных массивов

## Алгоритм решения

Создать метод transform, который будет принимать: cписок значений типа T, объект Function<T, P>, который будет применяться ко всем элементам списка. Для каждого элемента списка вызвать метод apply объекта Function и добавить результат в новый список. Вернуть новый список.

## Тестирование



## Код программы

[https://github.com/NTHrdd/Lab4](https://github.com/NTHrdd/Lab3)

# Задание 5

## Текст задания

Фильтр.

Разработайте такой метод, который будет принимать список значений типа T и объект имеющий единственный метод test (принимает T и возвращает boolean). Верните новый список типа T, из которого удалены все значения не прошедшие проверку условием. Используйте разработанный метод следующим образом:

1. Передайте в метод список со значениями: “qwerty”, “asdfg”, “zx”, и отфильтруйте все строки имеющие менее трех символов

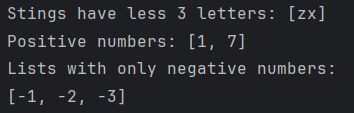
2. Передайте в метод список со значениями: 1,-3,7, и отфильтруйте все положительные элементы

3. Передайте в метод список состоящий из массивов целых чисел, а получите список в котором будут только те массивы, в которых нет ни одного положительного элемента

## Алгоритм решения

Создать метод filter, который будет принимать: список значений типа T, объект Predicate<T>, который будет проверять каждый элемент списка. Для каждого элемента вызвать метод test объекта Predicate. Если результат true, добавить элемент в новый список. Вернуть новый список.

## Тестирование



## Код программы

[https://github.com/NTHrdd/Lab4](https://github.com/NTHrdd/Lab3)

# Задание 6

## Текст задания

Сокращение.

Разработайте такой метод, который будет принимать список значений типа T и способ с помощью которого список значений можно свести к одному значению типа T, которое и возвращается из метода. Используйте разработанный метод следующим образом:

1. Передайте в метод список со значениями: “qwerty”, “asdfg”, “zx”, и сформируйте одну большую строку, которая состоит из всех строк исходного списка.

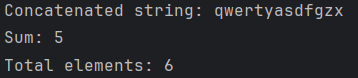
2. Передайте в метод список со значениями: 1,-3,7, и верните сумму всех значений исходного списка.

3. Имеется список, состоящий из списков целых чисел, получите общеe количество элементов во всех списках. Подсказка: решить задачу можно в одно действие или последовательно использовать методы из 3.3.1 и 3.3.3. Далее необходимо изменить разработанный метод таким образом, чтобы данный метод гарантированно не возвращал null и не выбрасывал ошибок в том случае, если исходный список пуст.

## Алгоритм решения

Создать метод reduce, который будет принимать: список значений типа T, функцию BiFunction<R, T, R>, которая объединяет два элемента в один. Начальное значение R. Для каждого элемента списка применить функцию BiFunction и обновить результат. Вернуть итоговое значение.

## Тестирование



## Код программы

[https://github.com/NTHrdd/Lab4](https://github.com/NTHrdd/Lab3)

# Задание 7

## Текст задания

Коллекционирование.

Разработайте такой метод, который будет возвращать коллекцию типа P со значениями типа T. Данный метод будет принимать:

1. Список исходных значений

2. Способ создания результирующей коллекции

3. Способ передачи значений исходного списка в результирующую коллекцию.

Используйте разработанный метод следующим образом:

1. Передайте в метод список со значениями: 1,-3,7, и верните их разбитыми на два подсписка, в одном из которых будут только положительные числа, а в другом только отрицательные.

2. Передайте в метод список со значениями: “qwerty”, “asdfg”, “zx”, “qw” и верните их разбитыми на подсписки таким образом, чтобы в любом подсписке были строки только одинаковой длины

3. Передайте в метод список со значениями: “qwerty”, “asdfg”, “qwerty”, “qw” и верните набор такого вида, который не может содержать одинаковые объекты.

## Алгоритм решения

Создать метод collect, который будет принимать: список значений типа T, фабрику коллекции Supplier<P>, функцию добавления элементов в коллекцию BiConsumer<P, T>. Для каждого элемента списка вызвать метод accept объекта BiConsumer, чтобы добавить элемент в коллекцию. Вернуть итоговую коллекцию.

## Тестирование

## Код программы

[https://github.com/NTHrdd/Lab4](https://github.com/NTHrdd/Lab3)