|  |  |  |
| --- | --- | --- |
| 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 5 | | |
|  | | Работу выполнил  студент группы ПМИ-1,2-2023 2 курса  Волегов И.С.  «16» декабря 2024 г. |
| Работу проверил  Ракина В.Д.  «\_\_» \_\_\_\_\_\_\_\_ 2024 г. |
| Пермь 2024 | | |

СОДЕРЖАНИЕ

[Задание 1 4](#_Toc185282922)

[Текст задания 4](#_Toc185282923)

[Алгоритм решения 4](#_Toc185282924)

[Тестирование 4](#_Toc185282925)

[Код программы 4](#_Toc185282926)

[Задание 2 5](#_Toc185282927)

[Текст задания 5](#_Toc185282928)

[Алгоритм решения 5](#_Toc185282929)

[Тестирование 5](#_Toc185282930)

[Код программы 5](#_Toc185282931)

[Задание 4 6](#_Toc185282932)

[Текст задания 6](#_Toc185282933)

[Алгоритм решения 6](#_Toc185282934)

[Тестирование 6](#_Toc185282935)

[Код программы 6](#_Toc185282936)

[Задание 5 6](#_Toc185282937)

[Текст задания 6](#_Toc185282938)

[Алгоритм решения 6](#_Toc185282939)

[Тестирование 7](#_Toc185282940)

[Код программы 7](#_Toc185282941)

[Задание 6 7](#_Toc185282942)

[Текст задания 7](#_Toc185282943)

[Алгоритм решения 7](#_Toc185282944)

[Тестирование 7](#_Toc185282945)

[Код программы 7](#_Toc185282946)

[Задание 7 8](#_Toc185282947)

[Текст задания 8](#_Toc185282948)

[Алгоритм решения 8](#_Toc185282949)

[Тестирование 8](#_Toc185282950)

[Код программы 8](#_Toc185282951)

[Инструкция по применению стилей и оформлению работы 8](#_Toc185282952)

# 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

Описание класса Box<T>:

Класс Box<T> реализует контейнер для хранения одного объекта типа T.

- Поля:

1. private T item;: хранит объект.

- Методы:

1. Box(): инициализирует item как null.

2. void put(T item):

- Добавляет элемент в коробку, если она пуста.

- Выбрасывает IllegalStateException, если уже заполнена.

3. T take():

- Извлекает элемент из коробки и очищает её.

4. boolean isFull():

- Возвращает true, если коробка заполнена.

5. String toString(): возвращает строковое представление содержимого коробки.

Метод main:

1. Создается объект Box<Integer> box.

2. Запрашивается ввод целого числа у пользователя с обработкой ошибок ввода.

3. Число помещается в коробку:

- Если успешно, выводится сообщение с состоянием коробки.

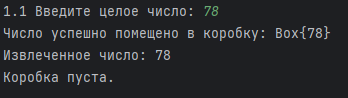
- Если коробка заполнена, отображается ошибка.

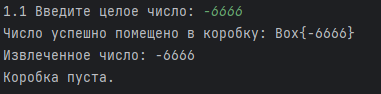
4. Извлекается число из коробки:

- Если есть значение, выводится оно; если нет — сообщение о пустой коробке.

5. Проверяется состояние коробки и выводится сообщение о её заполненности.

## Тестирование









## Код программы

Ссылка на githab

# Задание 2

## Текст задания

Без null. Создайте сущность Хранилище, которая обладает следующими характеристиками:

• Может хранить один произвольный объект в один момент времени. • Хранилище неизменяемо.

• Объект кладется в Хранилище при его создании. В качестве объекта может быть сохранено также и значение null.

• Хранилище может вернуть ссылку на Объект.

• Если вместо объекта хранится null, необходимо вернуть какое-либо альтернативное значение.

• Метод получения значения должен работать с тем типом данных, который был указан во время создания объекта Выполните следующие задания:

• Создайте Хранилище чисел, положите туда значение null. Передайте Хранилище в какойлибо метод, извлеките значение, и выведите его на экран. Альтернативой должно быть число 0.

• Создайте Хранилище чисел, положите туда значение 99. Передайте Хранилище в какойлибо метод, извлеките значение, и выведите его на экран. Альтернативой должно быть число -1.

• Создайте Хранилище строк, положите туда значение null. Передайте Хранилище в какойлибо метод, извлеките значение, и выведите его на экран. Альтернативой должна быть строка “default”.

• Создайте Хранилище строк, положите туда значение “hello”. Передайте Хранилище в какой-либо метод, извлеките значение, и выведите его на экран. Альтернативой должна быть строка “hello world”.

## Алгоритм решения

1. Класс Storage<T>:

- Этот класс представляет собой контейнер для хранения одного объекта типа T.

- Конструктор инициализирует объект значением, переданным в качестве параметра.

- Метод getValueOrDefault возвращает хранимое значение, если оно не равно null. Если значение равно null, возвращается значение по умолчанию, переданное в параметре.

- Метод toString выводит строковое представление содержимого контейнера, показывая либо сохраненное значение, либо "null", если оно отсутствует.

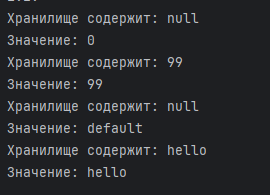
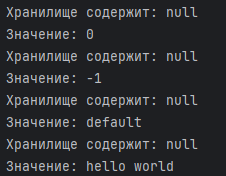
2. Main:

- Создаются объекты класса Storage<T> для хранения разных типов данных.

- Для каждого объекта выводится его строковое представление с помощью метода toString().

- Вызов метода getValueOrDefault для каждого объекта показывает либо хранимое значение, либо значение по умолчанию.

## Тестирование

## Код программы

Ссылка на githab

# 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

1. Метод universal:

* Принимает список элементов типа T и функцию преобразования Function<T, P>.
* Создает новый список resultList.
* Проходит по каждому элементу входного списка, применяет функцию преобразования и добавляет результат в resultList.
* Возвращает resultList.

1. Главный метод (main):

* Предлагает пользователю выбрать действие:

a) Преобразовать строки в их длины.

b) Сделать отрицательные числа положительными.

c) Найти максимальные элементы массивов.

- Читает выбор пользователя и обрабатывает его с помощью конструкции switch.

Действия в зависимости от выбора:

Выбор 1:

* Инициализирует список строк.
* Вызывает метод universal, передавая строковый список и метод для получения длин строк (String::length).
* Выводит длины строк.

Выбор 2:

* Инициализирует список чисел.
* Вызывает метод universal, передавая числовой список и метод для получения абсолютных значений (Math::abs).
* Выводит положительные числа.

Выбор 3:

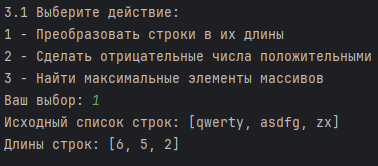
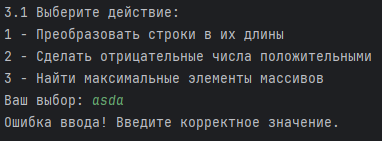
* Инициализирует список массивов.
* Вызывает метод universal, передавая список массивов и лямбда-функцию для нахождения максимального значения.
* Выводит максимальные значения из массивов.

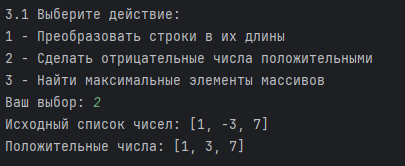
Обработка ошибок:

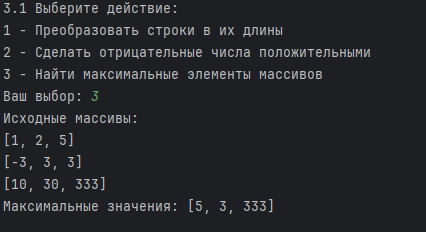
В случае неверного ввода выводится сообщение об ошибке.

.

## Тестирование





## Код программы

Ссылка на githab

# 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

1. Метод filter:

* Принимает список элементов типа T и предикат Predicate<T>, который определяет условие фильтрации.
* Создает новый список filteredList.
* Перебирает элементы списка и добавляет в filteredList только те элементы, которые удовлетворяют условию предиката.
* Возвращает отфильтрованный список.

1. Главный метод (main):

* Предлагает пользователю выбрать пример для тестирования:

1. Фильтрация строк длиной меньше 3 символов.
2. Фильтрация отрицательных чисел (оставляем только положительные).
3. Фильтрация массивов, содержащих только отрицательные числа.

* В зависимости от выбора пользователя выполняется соответствующая фильтрация.

Действия в зависимости от выбора:

Выбор 1:

* Инициализируется список строк.
* Вызвается метод filter, чтобы оставить только те строки, длина которых не меньше 3 символов.
* Выводится исходный и отфильтрованный список строк.

Выбор 2:

* Инициализируется список чисел.
* Вызвается метод filter, чтобы оставить только те числа, которые меньше или равны нулю (отфильтровываются положительные числа).
* Выводится исходный и отфильтрованный список чисел.

Выбор 3:

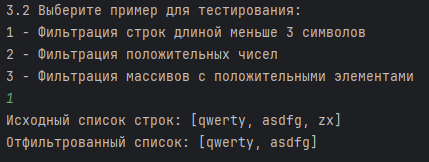
* Инициализируется список массивов целых чисел.
* Вызвается метод filter, чтобы оставить только те массивы, в которых все элементы меньше или равны нулю (отфильтровываются массивы с положительными числами).
* Выводится исходный и отфильтрованный список массивов.

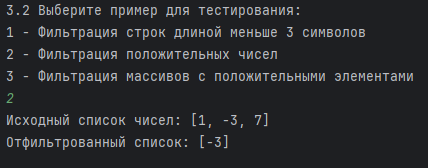
Обработка ошибок:

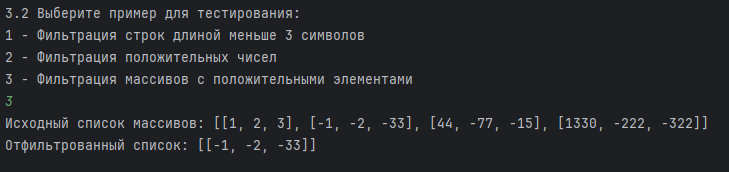
* Если пользователь выбирает неверный вариант, выводится сообщение о неправильном выборе.

.

## Тестирование







## Код программы

Ссылка на githab

# 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

1. Метод reduce:

* Принимает список элементов типа T, редуцирующую функцию Reducer<R, T>, и начальное значение типа R.
* Инициализирует результат значением initialValue.
* Проходит по каждому элементу списка, применяя редуцирующую функцию и обновляя результат.
* Возвращает итоговое значение результата после применения всех элементов списка.

1. Интерфейс Reducer:

* Определяет метод reduce, который принимает два параметра:

- celoe (текущее состояние результата).

- tek (текущий элемент списка).

- Метод возвращает обновленное значение результата после применения операции к элементу списка.

3.Объединение строк:

- Используется список: List<String> strings = List.of("qwerty", "asdfg", "zx").

- Применяется reduce с лямбда-функцией (a, b) -> a + b, чтобы объединить строки.

4. Суммирование чисел:

- Используется список: List<Integer> numbers = List.of(1, -3, 7).

- Применяется reduce с методом Integer::sum для сложения чисел.

## Тестирование

## 

## Код программы

Ссылка на githab

# 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

1. Метод collection:

* Параметры:

a) list: исходный список элементов типа T.

b) collectionCreator: функция для создания коллекции.

c) listProcessor: функция для обработки списка и получения результата.

- Процесс:

a) Метод вызывает функцию listProcessor.apply(list), которая обрабатывает переданный список и возвращает результат.

b) Результат обработки зависит от переданной функции listProcessor. В примерах выше это могут быть операции группировки, разделения или удаления дубликатов.

- Методы:

- collectionCreator: используется для создания коллекции (например, HashMap или HashSet).

- listProcessor: выполняет обработку списка и применяет одну из операций: разделение, группировка или создание множества уникальных значений.

.

## Тестирование

## 

## Код программы

Ссылка на githab