Министерство науки и высшего образования Российской Федерации

**Федеральное государственное бюджетное образовательное учреждение высшего образования**

«Владимирский государственный университет

имени Александра Григорьевича и Николая Григорьевича Столетовых»

**(ВлГУ)**

**Кафедра информационных систем и программной инженерии**

Лабораторная работа №3

по дисциплине «Платформонезависимое

программирование»

Тема: «Списочные структуры.»

Выполнил:

студент гр. ПРИ-123

А.Ц. Нямаа

Приняла:

Ст. пр. кафедры ИСПИ

О.Н. Шамышева

Владимир, 2025 г.

ЦЕЛЬ РАБОТЫ:

Познакомиться с представлением списочных структур и базовыми операциями над ними.

ЗАДАНИЕ 1 (вариант 12).

Выполнить пример из методического указания, заданием которого является вернуть сумму квадратов всех нечетных чисел в списке целых чисел A1,…,AN.

ВЫПОЛНЕНИЕ РАБОТЫ:

В начале выполнения данного задания я определил физическую структуру приложения-примера.

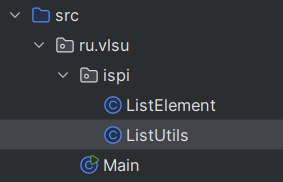


Рисунок 1. Физическая структура программы-примера.

В ходе реализации примера из методического указания создал несколько классов: исполняющий класс main, класс ListElement, представляющий собой элемент списка, и класс ListUtils, который является специальной библиотекой для работы со списком.

Листинг Main.java:

package ru.vlsu;  
  
import ru.vlsu.ispi.ListElement;  
import ru.vlsu.ispi.ListUtils;  
  
import java.util.Scanner;  
  
public class Main {  
 public static void main(String[] args) {  
 Scanner sc = new Scanner(System.in);  
 System.out.println("Введите количество элементов: ");  
 int N = sc.nextInt();  
 System.out.println("Введите значения элементов");  
 ListElement first = null;  
 for (int i = 0; i < N; i++) {  
 int value = sc.nextInt();  
 first = ListUtils.insert(first, value);  
 }  
 int result = ListUtils.reduce(ListUtils.map(ListUtils.filter(first)));  
 System.out.println(result);  
 }  
}

Листинг ListElement.java:

package ru.vlsu.ispi;  
  
public class ListElement {  
 private final int value;  
 private ListElement next;  
 public ListElement(int value)  
 {  
 this.value = value;  
 }  
 public int getValue() {  
 return value;  
 }  
 public ListElement getNext() {  
 return next;  
 }  
 public void setNext(ListElement next) {  
 this.next = next;  
 }  
}

Листинг ListUtils.java:

package ru.vlsu.ispi;  
  
public class ListUtils {  
 public static ListElement insert(ListElement first, int value) {  
 if (first == null) {  
 return new ListElement(value);  
 } else {  
 ListElement p = first;  
 while (p.getNext() != null)  
 p = p.getNext();  
 p.setNext(new ListElement(value));  
 return first;  
 }  
 }  
 public static ListElement filter(ListElement first) {  
 ListElement p = null;  
 while (first != null) {  
 if (first.getValue()%2 != 0) p = insert(p, first.getValue());  
 first = first.getNext();  
 }  
 return p;  
 }  
 public static ListElement map(ListElement first) {  
 ListElement p = null;  
 while (first != null) {  
 int value = first.getValue();  
 p = insert(p, value\*value);  
 first = first.getNext();  
 }  
 return p;  
 }  
 public static int reduce(ListElement first) {  
 int sum = 0;  
 while (first != null) {  
 sum += first.getValue();  
 first = first.getNext();  
 }  
 return sum;  
 }  
}

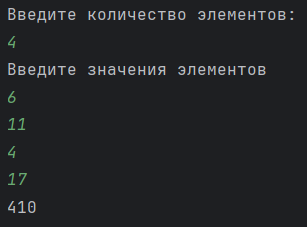


Рисунок 2. Выполнение программы-примера.

ЗАДАНИЕ 2 (вариант 12).

Необходимо запросить у пользователя число N и N элементов списка, после чего реализовать требуемый набор функций и вывести результат. Одну из функций необходимо реализовать рекурсивно.

Наборы функций:

Необходимо вывести пользователю сумму кубов всех элементов списка, которые делятся на три без остатка. Операцию вставки реализовать так, чтобы элементы добавлялись в начало списка, а не в конец. Функцию сопоставления (map) реализовать рекурсивно.

ВЫПОЛНЕНИЕ РАБОТЫ:

Для выполнения данного задания я определил физическую структуру приложения.

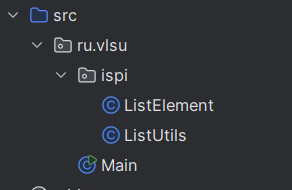


Рисунок 3. Физическая структура программы-примера.

Следующим пунктом написал код программы, реализовав некоторые классы (схожие с классами из программы-примера), но изменив набор функций на необходимый в соответствии с условием задачи.

Листинг Main.java:

package ru.vlsu;  
  
import ru.vlsu.ispi.ListElement;  
import ru.vlsu.ispi.ListUtils;  
  
import java.util.Scanner;  
  
public class Main {  
 public static void main(String[] args) {  
 Scanner sc = new Scanner(System.in);  
 System.out.println("Введите количество элементов: ");  
 int N = sc.nextInt();  
 System.out.println("Введите значения элементов: ");  
 ListElement first = null;  
 for (int i = 0; i < N; i++) {  
 int value = sc.nextInt();  
 first = ListUtils.insert(first, value);  
 }  
  
 System.out.println("Вывод списка: ");  
 ListElement current = first;  
 while (current != null){  
 System.out.print(current.getValue() + " ");  
 current = current.getNext();  
 }  
  
 System.out.println("\nСумма подходящих под условие элементов: ");  
 int result = ListUtils.reduce(ListUtils.map(ListUtils.filter(first), null));  
 System.out.println(result);  
 }  
}

Листинг ListElement.java:

package ru.vlsu.ispi;  
  
public class ListElement {  
 private final int value;  
 private ListElement next;  
 public ListElement(int value) {  
 this.value = value;  
 }  
 public int getValue() {  
 return value;  
 }  
 public ListElement getNext() {  
 return next;  
 }  
 public void setNext(ListElement next) {  
 this.next = next;  
 }  
}

Листинг ListUnits.java:

package ru.vlsu.ispi;  
  
public class ListUtils {  
 public static ListElement insert(ListElement first, int value) {  
 ListElement newElement = new ListElement(value);  
 newElement.setNext(first);  
 return newElement;  
 }  
 public static ListElement filter(ListElement first) {  
 ListElement p = null;  
 while (first != null) {  
 if (first.getValue() % 3 == 0){  
 p = insert(p, first.getValue());  
 }  
 first = first.getNext();  
 }  
 return p;  
 }  
 public static ListElement map(ListElement first, ListElement result) {  
 if (first == null){  
 return result;  
 }  
  
 result = insert(result, (int) Math.pow(first.getValue(), 3));  
 return map(first.getNext(), result);  
 }  
 public static int reduce(ListElement first) {  
 int sum = 0;  
 while (first != null) {  
 sum += first.getValue();  
 first = first.getNext();  
 }  
 return sum;  
 }  
}

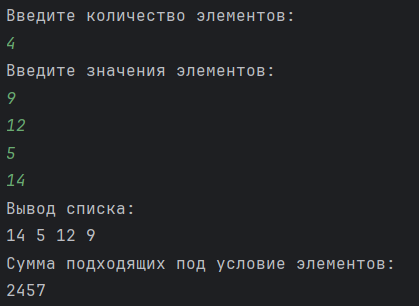


Рисунок 4. Выполнение программы-примера.

ВЫВОД К РАБОТЕ:

Я познакомился с представлением списочных структур и базовыми операциями над ними.