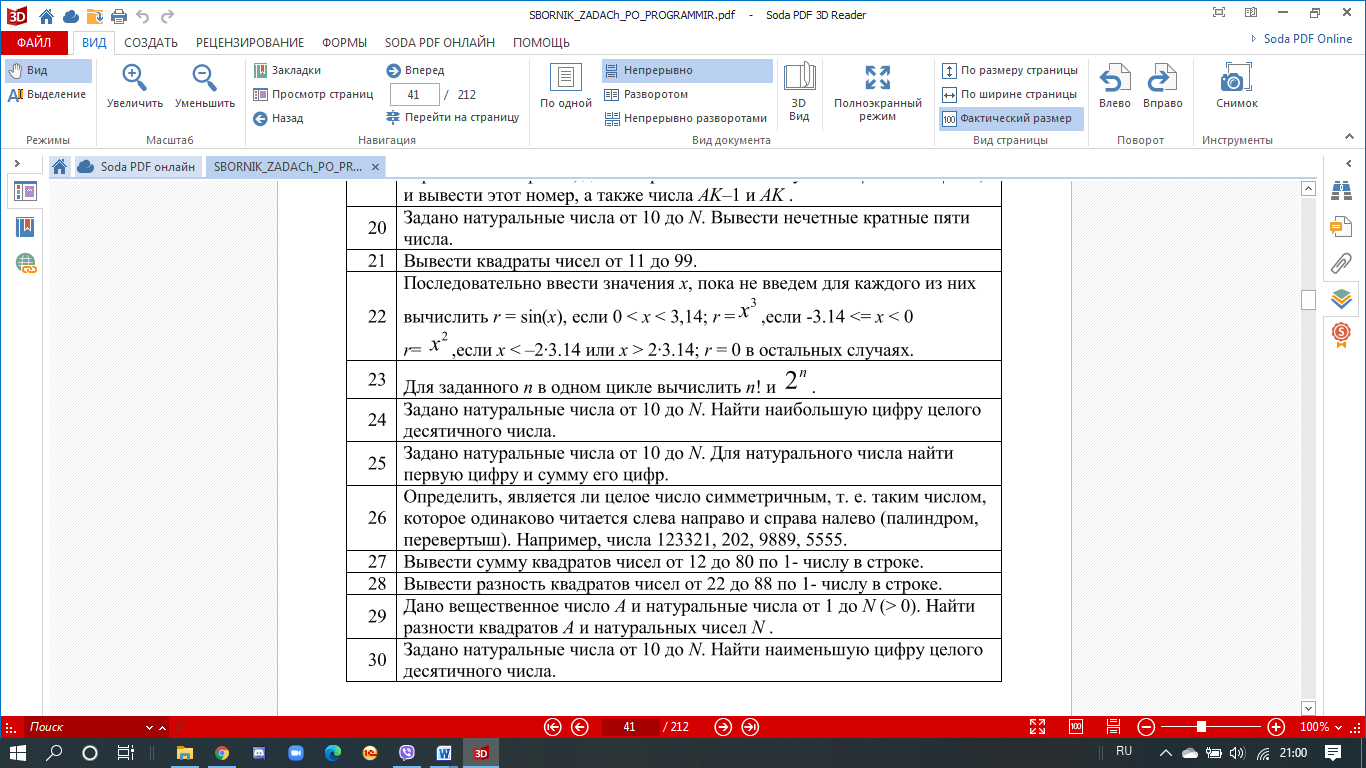
**Лабораторная работа №10**

**Основы программирования на Java: оператор цикла с параметром**

**Цель работы:** Освоить основные способы создания Java-программ либо с помощью обычного редактора, либо с помощью среды разработки. Приобрести навыки программирования циклических алгоритмов в Java.

Вариант 29:



Блок схема:

Начало

A, N

i = 1; i <= N; i++

difference=Разность квадратов A и N

Проверка N(=<0)

difference

Ошибка

Конец

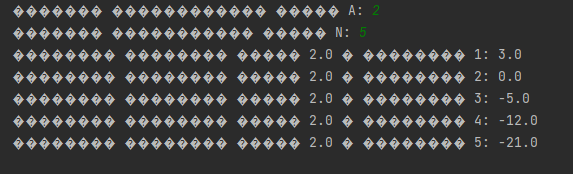
Да

Нет

|  |  |  |
| --- | --- | --- |
| Наименование переменной | Тип данных | Назначение |
| A | double | входные |
| N | int | входные |
| difference | double | выходные |

Код:

import java.util.Scanner;  
  
public class Main {  
 public static void main(String[] args) {  
 Scanner scanner = new Scanner(System.*in*);  
  
 System.*out*.print("Введите вещественное число A: ");  
 double A = scanner.nextDouble();  
  
 System.*out*.print("Введите натуральное число N: ");  
 int N = scanner.nextInt();  
  
 if (N <= 0) {  
 System.*out*.println("N должно быть натуральным числом больше 0.");  
 return;  
 }  
  
 for (int i = 1; i <= N; i++) {  
 double difference = Math.*pow*(A, 2) - Math.*pow*(i, 2);  
 System.*out*.println("Разность квадрата числа " + A + " и квадрата " + i + ": " + difference);  
 }  
 }  
}

Результат:

**Контрольные вопросы:**

1. **Цикл**:
   * Цикл в программировании представляет собой механизм, позволяющий повторять выполнение определенного блока кода многократно.
   * Он используется для автоматизации повторяющихся задач, таких как обработка массивов, выполнение итераций по коллекциям и других сценариев, требующих многократного выполнения кода.
2. **Операторы цикла в Java**:
   * **for**: предоставляет удобный способ выполнения кода определенное количество раз с явной инициализацией, условием и выражением итерации.
   * **while**: выполняет блок кода, пока условие остается истинным.
   * **do-while**: Похож на **while**, но гарантирует выполнение блока кода хотя бы один раз.
   * Усовершенствованный цикл **for-each**: упрощает итерацию по элементам массивов и коллекций.
3. **Оператор for**:
   * **for** в Java имеет три части: инициализация, условие и выражение итерации.
   * Он позволяет более компактно выражать циклические операции и удобен, когда известно количество итераций.
4. **Улучшенный цикл for** (for-each):
   * Предоставляет удобный способ итерации по элементам массивов и коллекций без явной работы с индексами.
   * Это делает код более читаемым и уменьшает вероятность ошибок.
5. **Операторы break, continue**:
   * **break**: используется для немедленного выхода из цикла.
   * **continue**: пропускает оставшуюся часть кода в текущей итерации и переходит к следующей итерации цикла.
   * **break** помогает прервать выполнение цикла, когда условие выполнено.
   * **continue** пропускает остаток кода в текущей итерации и начинает следующую.

Начало формы