**Лабораторная работа № 3**

«**Операторы управления выполнением программы и**

**работа с классами**»

по курсу: «**Введение в программирование на Java**»

**Цель работы**: Изучить особенности построения Java-программ на основе ООП. Изучить работу с операторами циклов и ветвления.

**Задание к лабораторной работе:**

1. Разработать в программе следующие классы:

- класс, содержащий функцию main;

- класс для методов основной программы (методы должны содержать решение задачи по номеру своего варианта; в основном методе должны быть использованы условия и циклы);

- класс для методов тестирования, производный от класса основной программы (методы тестирования должны реализовывать тестирование методов основной программы; в методах тестирования должны быть протестированы все крайние случаи работы программы).

2. Создать объекты классов программы и тестирования в функции main().

3. Выполнить и протестировать программу.

**Содержание отчёта:**

1. Титульный лист.

2. Условие задачи.

3. Ручной расчёт результата работы программы для всех тестов.

4. Распечатка программы.

5. Экранные формы.

**Варианты заданий:**

1. Спортсмен пробегает за 1-й день М км, каждый последующий день он увеличивает норму пробега на К%. Определите, через сколько дней норма пробега может стать более 50 км.

2. Стоимость акций известной фирмы стабильно растёт на протяжении 6 месяцев, каждый месяц на 0,6%. Найдите среднюю стоимость акций за эти полгода, если известно, что начальная стоимость их была К$.

3. Винни-Пух на каждый свой день рождения съедает столько еды, сколько съел на двух предыдущих вместе. На первых двух днях он съел по M граммов. Нужно узнать, сколь граммов еды он съест на N-й день рождения...

4. В бригаде, работающей на уборке сена, имеется N косилок. Первая из них работала M часов, а каждая следующая на 10 минут больше, чем предыдущая. Сколько часов проработала вся бригада?

5. Каждая бактерия делится на две в течение одной минуты. В начальный момент имеется одна бактерия. Составьте программу, которая рассчитывает количество бактерий на заданное вами целое значение момента времени (15 минут, 7 минут и т.п.).

6. В 1202г. Итальянский математик Леонард Пизанский (Фибоначчи) предложил такую задачу: пара кроликов каждый месяц даёт приплод – двух кроликов (самца и самку), от которых через два месяца уже получается новый приплод, Сколько кроликов будет через N месяцев, если в начале года имелась одна пара?

7. Найти количество таких четырёхзначных чисел в интервале [a..b], у которых все цифры различны.

8. Натуральное число из n цифр является число Армстронга, если сумма его цифр возведенных в n-ую степень равна самому числу. Нужно найти наименьшее такое число, большее заданного X.

9. Посчитать (a^b) mod c.

10. Поменять порядок элементов в массиве на обратный.

11. Отсортировать массив по возрастанию любым методом.

12. Найти максимум среди всех чётных по значению элементов массива.

13. Найти среднее всех положительных элементов массива.

14. Найти сумму всех чётных по значению элементов массива.

15. Посчитать a^b.

16. Определить количество дней между двумя датами (с учетом разных лет от 0..9999).

17. Найти наименьший из элементов, расположенных правее максимального элемента массива.

Пример: массив 7 2 6 5 4 4 10 3 9

максимальный элемент 10

наименьший правее максимального 3

18. После каждого четного элемента массива вставить два элемента с тем же значением.

Пример: массив 7 2 6 5 4 4 10 3 9

четные элементы 2 6 4 4 10

массив после вставки

7 2 2 2 6 6 6 5 4 4 4 4 4 4 10 10 10 3 9

19. Напишите функцию, которая находит в массиве элемент, значение которого равно x. Функция возвращает индекс найденного элемента или -1, если такого элемента в массиве нет.

20. Напишите функцию, которая подсчитывает, сколько раз в массиве встречается значение, являющееся максимальным. Функция должна выполнять однократный просмотр массива.

21. Даны два отсортированных массива: int A[n] и int B[m].   
Объедините их в один отсортированный массив int C[n+m], то есть если A={1, 4, 6, 7}, B={2, 3, 5}, то C={1, 2, 3, 4, 5, 6, 7}. Оформите алгоритм в виде функции void merge (int A[], int n, int B[], int m, int C[]). Время работы алгоритма должно быть порядка n+m действий. Пример:   
          Вход

          1 5 7

           2 4 4 5  
          Выход  
          1 2 4 4 5 5 7

22. Создать массив В из четных элементов массива А.

Пример: массив А 7 2 6 5 4 4 10 3 9

массив В 2 6 4 4 10

23.Сосчитать количество единиц в двоичной записи числа i.

Пример: массив А 1 2 3 4 5 6, m=4, n=2

массив В 5 6 1 2 3 4

24. Дан массив целых чисел x[1]..x[m+n], рассматриваемый как соединение двух его отрезков: начала x[1]..x[m] длины m и конца x[m+1]..x[m+n] длины n. Не используя дополнительных массивов, переставить начало и конец. (Число действий порядка m+n.)

25. Массив длины N заполнен в случайном порядке числами из диапазона от 1 до k < N. Не используя других массивов, подсчитать количество различных чисел. Число действий должно быть порядка N+k.

26. Вставить в упорядоченный по возрастанию массив новый элемент таким образом, сохранилась упорядоченность.

27. В массив В записать все элементы массива А, встречающиеся больше одного раза.

Пример: массив А 7 3 6 5 4 3 10 3 4

элементы встречающиеся

больше одного раза 3 и 4

массив В 3 4

28. Заданы массивы А и В. В массив С переписать все элементы массива А, которые встречаются в массиве В не меньше чем 2 раза.

Пример: массив А 7 3 7 5 4 3

массив В 1 3 4 2 3 6 4 1 3

массив С 3 4

29. Заданы массивы А, В и С. Из массива А удалить элементы, совпадающие по значению с теми из элементов массива В, которых нет в массиве С.

Пример: массив А 7 3 7 5 4 9

массив В 1 3 4 2 3 6 4 1 3 5

массив С 3 1 8 9

из массива А не удаляются:

7 - нет в В

3 - есть в В, но есть и в С

9 - нет в В

из массива А удаляются:

5 - есть в В и нет в С

4 - есть в В и нет в С

массив А после обработки 7 3 7 9

A=A\(B\C)

30. Заданы массивы А и В. Из массива А удалить все элементы, которые встречаются в В ровно 2 раза. Из В переписать в С те элементы, которые совпадают с оставшимися в массиве А. Выходной массив - С.

Пример: массив А 7 3 7 5 4 3

массив В 1 3 4 2 3 1 4 7 3

в массиве В 2 раза встречаются 1 4

массив А после удаления 7 3 7 5 3

элементы В совпадающие с А 3 3 7 3

массив С 3 3 7 3

массив В 1 4 2 6 4