

Модуль 1

Задание 1

Напишите приложение для решения квадратного уравнения вида:

$$ax^2+bx+c=0$$

Весь код Вашего приложения разместите в **public static void main(String[] args)**

Вывод решения на экран реализуйте с использованием **System.out.println()**

Определите переменные типа **double** для описания переменных **a**, **b**, **c**

```
double a = 1;
```

```
double b = -8;
```

```
double c = 12;
```

Проверьте поведение программы для случаев когда:

- `a == 0`
- `a == 5, b == 0, c == 30`

Для вычисления квадратного корня используйте конструкцию **Math.sqrt()**, например, **Math.sqrt(4)**

Модуль 2

Задание 1

С использованием циклов напишите приложение, которое выводит на экран текст вида

```
1*****
12*****
123*****
1234***
12345**
123456*
1234567
```

Для вывода на экран с переносом строки используйте `System.out.println()`

Для вывода на экран без переноса строки используйте `System.out.print()`

Задание 2

Отсортируйте массив из 10 элементов (целых чисел) с использованием алгоритма пузырьковой сортировки (https://en.wikipedia.org/wiki/Bubble_sort)

Выведите исходный массив и отсортированный массивы на экран

Исходный массив Вы можете задать или статически (на этапе инициализации) или динамически с использованием конструкции генератора случайных чисел `Random`:

```
//Create random number generator
Random rn = new Random();
//get new random integer
int a = rn.nextInt(10);
```

Задание 4

Реализуйте алгоритм бинарного поиска в отсортированном массиве (https://en.wikipedia.org/wiki/Binary_search_algorithm)

Задание 5

Напишите приложение, которое выводит на экран «Odd number» в случае, если заданное число нечетное и «Even number» если число четное.

Для задания исходного числа определите переменную `number`

Задание 6

Напишите приложение, которое выводит текстовое представление числа от одного до девяти.

Например, если исходное число 6, то на экране должно вывестись «Six»

Для определения исходного числа определите переменную `number`

Задание 7

Напишите приложение, которое выводит на экран сумму всех чисел от 1 до 100. Также выведите на экран среднее этих чисел.

Задание 6

Напишите приложение которое вычисляет число π до 10 знака. Сравните полученный результат с константой Math.PI

Алгоритм вычисления числа π произвольный

Задание 7

Напишите приложение которое выводит на экран сумму ряда

$$1 + \frac{1}{2} + \frac{1}{3} + \frac{1}{4} + \frac{1}{5} + \dots + \frac{1}{n}$$

где $n = 50000$.

Модуль 3

Задание 1

Напишите приложение калькулятор состоящие из двух классов:
первый класс выполняет базовые операции такие как сложение, вычитание, умножение и деление
второй класс использует первый

Методы сложения, вычитания, деления и умножения должны быть статическими

Дополните первый класс методами реализующими память: добавление в память, вычитание из памяти и очистка памяти (аналоги кнопок M+, M- и MR в обычном калькуляторе)

Методы для работы с памятью должны быть НЕ статическими

Задание 2

Напишите приложение калькулятор состоящие из двух классов:

Первый класс Person с полями name, surname и email.

- Напишите два конструктора: один конструктор по умолчанию, второй с аргументами name, surname, email
- Напишите метод для вывода информации класса Person на экран
- Напишите методы для доступа к полям класса Person используя понятия геттер и сеттер (<https://dzone.com/articles/why-should-i-write-getters-and-setters>)

Второй класс должен создать два экземпляра класса Person и, используя метод вывода на экран класса Person, вывести имя, фамилию и email

Для полей name, surname и email используйте тип String

Задание 3

Создайте класс описывающий односвязный (Singly linked list) список (https://en.wikipedia.org/wiki/Linked_list).
Предусмотрите методы добавления в список, удаления из списка и получения элемента списка по индексу

Превратите класс Вашего связного списка в двусвязный (Doubly linked list)

Задание 4

Создайте класс описывающий бинарное дерево (Binary tree — https://en.wikipedia.org/wiki/Binary_tree). В качестве данных в вершинах должны быть целые числа
Предусмотрите возможность добавления элементов в любую из вершин.

Используя алгоритм поиска в глубину выполните поиск заданного числа в дереве