

Лабораторна робота №1

Структура програми мовою Java. Типи даних, літерали, операції і оператори

Мета: Ознайомлення з *JDK* платформи *Java SE* та середовищем розробки *Eclipse IDE*.

1. Вимоги:

1. Вирішити три прикладні задачі на мові *Java* в середовищі *Eclipse*
2. Продемонструвати покрокове виконання програми та результати роботи в режимі налагодження, не використовуючи виведення до консолі.
3. Виконати компіляцію і запуск програми в командному рядку за допомогою відповідних утиліт *JDK*.

1.1. Розробник:

- Кедровський Максим
- КІТ-119а
- 10 варіант

1.2. Загальне завдання:

1. Обрати тип змінних та встановити за допомогою констант та літералів початкові значення:
 - 1.1. число, що відповідає номеру залікової книжки за допомогою шістнадцяткового літералу;
 - 1.2. число, що відповідає номеру мобільного телефона (починаючи з 380...) за допомогою десяткового літералу;
 - 1.3. число, яке складається з останніх двох ненульових цифр номера мобільного телефону за допомогою двійкового літералу;
 - 1.4. число, яке складається з останніх чотирьох ненульових цифр номера мобільного телефону за допомогою вісімкового літералу;
 - 1.5. визначити збільшене на одиницю значення залишку від ділення на 26 зменшеного на одиницю номера студента в журналі групи;
 - 1.6. символ англійського алфавіту в верхньому регістрі, номер якого відповідає знайденому раніше значенню.
2. Використовуючи десятковий запис цілочисельного значення кожної змінної знайти і підрахувати кількість парних і непарних цифр.

3. Використовуючи двійковий запис цілочисельного значення кожної змінної підрахувати кількість одиниць.

2. Опис програми:

```
1 package task01;
2
3 public class Main {
4     static int a = 0x8; //число, що відповідає номеру залікової книжки за допомогою шістнадцяткового літералу;
5     static long b = 3806606375681; //число, що відповідає номеру мобільного телефону (починаючи з 380...) за допомогою десяткового літералу;
6     static byte c = 0b1000100; //число, яке складається з останніх двох ненульових цифр номера мобільного телефону за допомогою двійкового літералу;
7     static int d = 016620; //число, яке складається з останніх чотирьох ненульових цифр номера мобільного телефону за допомогою вісімкового літералу;
8     static int e = (a-1)%26 + 1; //визначити збільшене на одиницю значення залишку від ділення на 26 зменшеного на одиницю номера студента в журналі групи;
9     static char f = (char) e; //символ англійського алфавіту в верхньому регістрі, номер якого відповідає знайденому раніше значенню.
10    static long[] arr;
11    public static void main(String[] args) {
12        arr = new long[6];
13        //Ворчунаю занову значення созданных ранее переменных в числовой массив
14        arr[0] = a;
15        arr[1] = b;
16        arr[2] = c;
17        arr[3] = d;
18        arr[4] = e;
19        arr[5] = f;
20        int evens = 0;
21        int odds = 0;
22
23        //Поиск четный и нечетных чисел в массиве
24        for(int i = 0; i < arr.length; i++) {
25            System.out.println(arr[i]);
26            if (arr[i]%2 == 0) evens++;
27            else odds++;
28        }
29        System.out.println("evens = " + evens + "\n" + "odds = " + odds);
30
31        //Поиск единиц в двоичном виде переменных
32        int ones = 0;
33        for(int i = 0; i < arr.length; i++) {
34            String s = Integer.toBinaryString((int)arr[i]); //Перевод чисел в двоичное представление(тип - строка)
35            //System.out.println(s);
36            for(int j = 0; j < s.length(); j++) { //Поиск в каждой строке единиц
37                if (s.charAt(j) == '1') ones++; //charAt - получение символа из строки по индексу
38            }
39        }
40        System.out.println("Ones in binary = " + ones);
41    }
42 }
```

Рисунок 1.1 - Код Main.java

Змінні a-f об'явлено за допомогою різних способів, у двійковому, вісімковому, десятковому виді, а також як символ. Завдання на пошук парних і непарних чисел виконано за допомогою циклу та операції залишок від ділення(%). Для полегшення завдання з роботи з двійковим форматом використано бібліотечні методи toBinaryString() для швидкого переходу до двійкового представлення та charAt() для почергового перегляду кожного символу строки.

3. Варіанти використання:

Програма призначена для демонстрації основних принципів роботи з мовою java, виконання простих операцій. Розроблений функціонал дозволяє шукати парні і непарні числа, а також рахувати кількість одиниць у двійковому записі чисел.

Висновок:

Ознайомився з *JDK* платформою *Java SE* та середовищем розробки *Eclipse IDE*. Мовою Java було розроблено програму відповідно до індивідуального завдання.