Звіт

Автор: Лунгол О., КІТ101.8б

Дата: 08.09.2019

Лабораторна робота №1

СТРУКТУРА ПРОГРАМИ МОВОЮ JAVA. ТИПИ ДАНИХ, ЛІТЕРАЛИ, ОПЕРАЦІЇ І ОПЕРАТОР

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

**Вимоги:**

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

ЗАВДАННЯ ДО РОБОТИ

1. Обрати тип змінних та встановити за допомогою констант та літералів початкові значення:

* число, що відповідає номеру залікової книжки за допомогою шістнадцяткового літералу;
* число, що відповідає номеру мобільного телефона (починаючи з 380…) за допомогою десяткового літералу;
* число, яке складається з останніх двох ненульових цифр номера мобільного телефону за допомогою двійкового літералу;
* число, яке складається з останніх чотирьох ненульових цифр номера мобільного телефону за допомогою вісімкового літералу;
* визначити збільшене на одиницю значення залишку від ділення на 26 зменшеного на одиницю номера студента в журналі групи;
* 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.

ОПИС ПРОГРАМИ

**Опис змінних**

**final** **short** Book; // номер залікової книжки

**final** **long** Phone; // номер мобільного телефону

**final** **byte** binaryLastNum; // дві останні ненульові цифри номеру

**final** **short** octalLastNum; // останні 4 ненульові цифри номеру

**final** **byte** numG; // порядковий номер у журналі групи

**final** **int** number; // зберігання результату формули

**final** **char** eng; // отримання з формули номер букви

**Ієрархія та структура класів**

**class** Main – точка входу в програму(Рисунок 1).

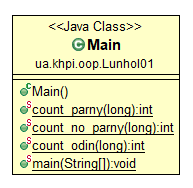


Рисунок 1 - **class** Main

ТЕКСТ ПРОГРАМИ

**package** ua.khpi.oop.Lunhol01;

**public** **class** Main {

**public** **static** **int** count\_parny(**long** number) {

**long** buf=number;

**long** num=0;

**int** count\_parny = 0;

**while**(buf != 0) {

num = buf % 10;

**if**(num % 2 == 0) {

count\_parny++;

}

buf /= 10;

}

**return** count\_parny;

}

**public** **static** **int** count\_no\_parny(**long** number) {

**long** buf=number;

**long** num=0;

**int** count\_no\_parny = 0;

**while**(buf != 0) {

num = buf % 10;

**if**(num % 2 != 0) {

count\_no\_parny++;

}

buf /= 10;

}

**return** count\_no\_parny;

}

**public** **static** **int** count\_odin(**long** one) {

String bin = Long.*toBinaryString*(one);

**int** count\_bin = 0;

**for**(**int** i = 0; i < bin.length(); i++) {

**if**(bin.charAt(i) == '1') {

count\_bin++;

}

}

**return** count\_bin;

}

**public** **static** **void** main(String[] args) {

**final** **short** Book=0x4665; //0х для считывания числа в 16 системе

**final** **long** Phone=380989568786L; //L для типа long

**final** **byte** binaryLastNum=0b1010110; //0b двоичный литерал

**final** **short** octalLastNum=021122; //0 в начале для восьмиричного литерала

**final** **byte** numG=10;

**final** **int** number=(numG-1)%26+1;

**final** **char** eng=(**char**)number+65;

System.***out***.println(eng);

System.***out***.println("Numer: " + Book + ", Parny: " + *count\_parny*(Book) + ", Not parny: " + *count\_no\_parny*(Book)+", BinaryNum: "+Long.*toBinaryString*(Book)+",BinaryCount: "+*count\_odin*(Book));

System.***out***.println("Numer: " + Phone + ", Parny: " + *count\_parny*(Phone) + ", Not parny: " + *count\_no\_parny*(Phone)+", BinaryNum: "+Long.*toBinaryString*(Phone)+",BinaryCount: "+*count\_odin*(Phone));

System.***out***.println("Numer: " + binaryLastNum + ", Parny: " + *count\_parny*(binaryLastNum) + ", Not parny: " + *count\_no\_parny*(binaryLastNum)+", BinaryNum: "+Long.*toBinaryString*(binaryLastNum)+",BinaryCount: "+*count\_odin*(binaryLastNum));

System.***out***.println("Numer: " + octalLastNum + ", Parny: " + *count\_parny*(octalLastNum) + ", Not parny: " + *count\_no\_parny*(octalLastNum)+", BinaryNum: "+Long.*toBinaryString*(octalLastNum)+",BinaryCount: "+*count\_odin*(octalLastNum));

System.***out***.println("Numer: " + numG + ", Parny: " + *count\_parny*(numG) + ", Not parny: " + *count\_no\_parny*(numG)+", BinaryNum: "+Long.*toBinaryString*(numG)+",BinaryCount: "+*count\_odin*(numG));

System.***out***.println("Numer: " + number + ", Parny: " + *count\_parny*(number) + ", Not parny: " + *count\_no\_parny*(number)+", BinaryNum: "+Long.*toBinaryString*(number)+",BinaryCount: "+*count\_odin*(number));

}

}

РОБОТА ПРОГРАМИ

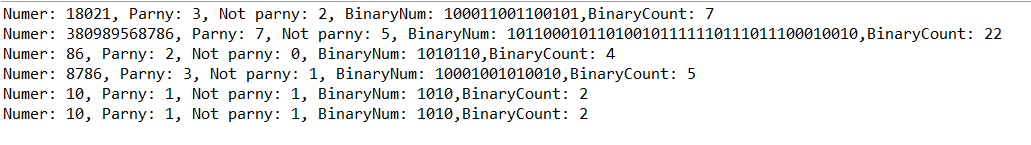


Рисунок 2 – Результат роботи програми

ВАРІАНТИ ВИКОРИСТАННЯ

Використати програму у корисних цілях скоріш за все не вдасться. Але можна використати схему знаходження кількості парних, непарних і взагалі будь-яких цифр у великій кількості чисел.

ВИСНОВКИ

При виконанні лабораторної роботи набуто практичних навичок, щодо розробки програм за допомогою JDK платформи Java SE. Програма виконується без помилок.