**ЛАБОРАТОРНА РОБОТА № 1**

**РОБОТА З БАЗОВИМИ ТИПАМИ ДАНИХ**

***Мета:*** ознайомитись з основними можливостями мови програмування Go, найпростішої структурою програми, з властивостями базових типів. Отримати практичні навички по роботі з базовими типами даних в мові Go.

**Хід роботи**

**Завдання 1**

*//1. Вывести текст на украинском языке*

    var strUkr string = "Привіт світ\nУкраїнська мова"

    fmt.Println(strUkr)

    strUkr1 := "Їжак"

    fmt.Println("-> " + strUkr1)

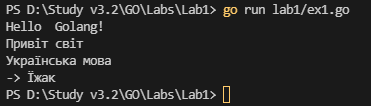


Рис.1. Результат

**Завдання 2**

*//Задание.*

*//1. Создать целочисленную переменную (результат не отображать)*

    userInt := 10

    var int32Value int32 = -15

**Завдання 3**

*//Задание.*

*//1. Вывести типы всех переменных*

    fmt.Printf("Variables types -> %T, %T, %T, %T\n", userinit8, userinit16, userinit64, userautoinit)

*//2. Присвоить переменной intVar переменные userinit16 и userautoinit. Результат вывести.*

    intVar = userautoinit

    fmt.Printf("intVar = %d\n", intVar)

    intVar = int(userinit16)

    fmt.Printf("intVar = %d\n", intVar)



Рис.2. Результат

**Завдання 4**

*//Задание.*

*//1. Выпонить выражение: fmt.Println("c--     = ", c--)*

    c--

    fmt.Println("\nc--     = ", c)

    fmt.Println("c--     = ", c-1)



Рис.3. Результат

**Завдання 5**

*//Задание.*

*//1. Определить разрядность ОС*

    var x uint32

    a := unsafe.Pointer(&x)

*if* unsafe.Sizeof(a) == 8 {

        fmt.Println("x64")

    } *else* *if* unsafe.Sizeof(a) == 4 {

        fmt.Println("x32")

    }

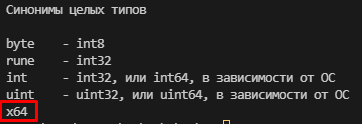


Рис.4. Результат

**Завдання 6**

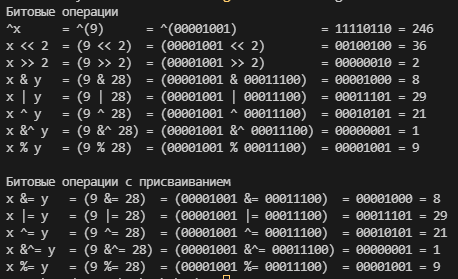


Рис.5. Бітові операції

*//Задание.*

*//1. Пояснить результаты операций*

*// ^ - поразрядное исключающее ИЛИ. Возвращает 1, если только один из соответствующих разрядов обоих чисел равен 1*

*//     0 0 0 0 1 0 0 1*

*//  ^  0 0 0 1 1 1 0 0*

*//  =  0 0 0 1 0 1 0 1*

*// << - cдвигает битовое представление числа, представленного первым операндом, влево на определенное количество разрядов, которое задается вторым операндом*

*// << 2 - на два разряда влево*

*// >> - Сдвигает битовое представление числа вправо на определенное количество разрядов*

*// >> 3 - на три разряда вправо*

*// & - поразрядная конъюнкция (операция И или поразрядное умножение)*

*// Возвращает 1, если оба из соответствующих разрядов обоих чисел равны 1. Возвращает 0, если разряд хотя бы одного числа равен 0*

*//    0 0 0 0 1 0 0 1*

*// &  0 0 0 1 1 1 0 0*

*// =  0 0 0 0 1 0 0 0*

*// | - поразрядная дизъюнкция (операция ИЛИ или поразрядное сложение)*

*// Возвращает 1, если хотя бы один из соответствующих разрядов обоих чисел равен 1*

*//    0 0 0 0 1 0 0 1*

*// |  0 0 0 1 1 1 0 0*

*// =  0 0 0 1 1 1 0 1*

*// &^ - сброс бита*

*// ^() - реверс бита*

*// % - возвращает остаток от деления*

**Завдання 7**

*/Задание.*

*//1. Создайте 2 переменные  разных типов. Выпоните арифметические операции. Результат вывести*

    var a int = 10

    var b float64 = -3.4

    fmt.Printf("\na + b = %d", a+int(b))

    fmt.Printf("\na + b = %d", a-int(b))

    fmt.Printf("\na + b = %d", a\*int(b))

    fmt.Printf("\na + b = %d", a/int(b))

    fmt.Printf("\na + b = %d", a%int(b))

    b++

    fmt.Printf("\nb++")

    fmt.Printf("\na + b = %d", a/int(b))

    a--

    fmt.Printf("\na--")

    fmt.Printf("\na + b = %d", a/int(b))

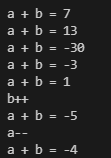


Рис.6. Результат

**Завдання 8**

*//Задание.*

*//1. Создайте переменные разных типов, используя краткую запись и инициализацию по-умолчанию. Результат вывести*

    a := 4

    b := 7.5

    fmt.Printf("\na = %d, b = %f", a, b)



Рис.7. Результат

**Завдання 9**

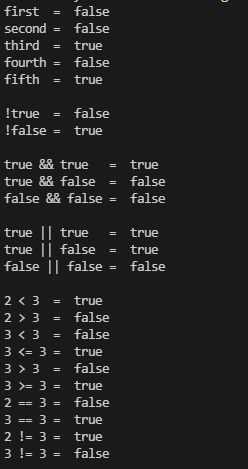


Рис.8. Результаты операций

*//Задание.*

*//1. Пояснить результаты операций*

*// && - возвращает true, если оба операнда не равны false. Возвращает false, если хотя бы один операнд равен false\*

*// || - Возвращает true, если хотя бы один операнд не равен false. Возвращает false, если оба операнда равны false*

**Завдання 10**

*//Задание.*

*//1. Вывести украинскую букву 'Ї'*

    fmt.Printf("ї\n")

*//2. Пояснить назначение типа "rune"*

*// тип данных для хранения одного символа в кодировке*

    var letter rune = 'Ї'

    fmt.Printf("%c - %d\n", letter, letter)

    fmt.Println(letter)



Рис.9. Результат

***Висновки:*** у ході виконання лабораторної роботи було ознайомлено з основними можливостями мови програмування Go, найпростішої структурою програми, з властивостями базових типів. Отримано практичні навички по роботі з базовими типами даних в мові Go.