# МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное автономное образовательное учреждение высшего образования

## «СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Кафедра инфокоммуникаций

«Основы языка программирования Go»

Отчет по лабораторной работе № 1 по дисциплине «Программная инженерия»

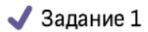
Выполнил студент группы ПИ	Ж-б-о-21-1
Гребенкин Е. А	«» 2022г.
Подпись студента_	
Работа защищена «»	20г.
Проверил Воронкин Р.А.	_ (подпись)

**Цель работы:** исследование назначения и способов установки Go, исследование типов данных, констант и арифметических операции языка программирования Go.

### Выполнение работы:

- 1. Изучить теоретический материал работы.
- 2. Создать общедоступный репозиторий на GitHub, в котором будет использована лицензия МІТ и язык программирования Python.
- 3. Выполните клонирование созданного репозитория.
- 4. Дополните файл .gitignore необходимыми правилами для работы с IDE.
- 5. Проработайте примеры теоретической части.
- 6. Решите все задания практической части.
- 7. Решите индивидуальное задание согласно варианту (номер варианта необходимо уточнить у преподавателя).
- 8. Используйте для каждой задачи (проекта) отдельную папку.
- 9. Добавьте отчет по лабораторной работе в формате PDF в папку doc репозитория. Зафиксируйте изменения.
- 10.Выполните слияние ветки для разработки с веткой master/main.
- 11.Отправьте сделанные изменения на сервер GitHub.
- 12. Отправьте адрес репозитория GitHub на электронный адрес преподавателя.

### Решение практических задач



Задача: Напишите программу, которая выводит "I like Go!"

Пример ввода:

Соответствующий вывод:

I like Go!

```
package main

import "fmt"

func main() {

    // Задание 1
    fmt.Println("I like Go!")

}
```

Листинг 1 – Код решения первой задачи

```
PS C:\Users\3C\Desktop\Учёба\Программная инженерия\Практики\1 paботa\PE_1\tasks> go run ./task1/main.go
I like Go!
PS C:\Users\3C\Desktop\Учёба\Программная инженерия\Практики\1 paботa\PE_1\tasks>
```

Рисунок 1 – Вывод программы первой задачи



Задача: Напишите программу, которая выведет "I like Go!" 3 раза.

Пример ввода:

```
I like Go!
I like Go!
I like Go!
```

```
package main
import "fmt"

func main() {
    // Задание 2
    for i := 0; i < 3; i++ {
        fmt.Println("I like Go!")
    }
}</pre>
```

Листинг 2 – Код решения второй задачи

```
PS C:\Users\3C\Desktop\Учёба\Программная инженерия\Практики\1 paбота\PE_1\tasks> go I like Go! I like Go! I like Go! I like Go! PS C:\Users\3C\Desktop\Учёба\Программная инженерия\Практики\1 paбота\PE_1\tasks>
```

Рисунок 2 – Вывод программы второй задачи

```
✓ Задание 3
```

**Задача:** Напишите программу, которая последовательно делает следующие операции с введённым числом:

- 1. Число умножается на 2;
- 2. Затем к числу прибавляется 100.

После этого должен быть вывод получившегося числа на экран.

Пример ввода:

```
1
```

```
102
```

```
package main
import "fmt"

func main() {

    // Задание 3
    var num int

    fmt.Scan(&num)
    fmt.Println(num*2 + 100)
}
```

Листинг 3 – Код решения третьей задачи

```
PS C:\Users\3C\Desktop\Учёба\Программная инженерия\Практики\1 paбота\PE_1\tasks> go
1
102
PS C:\Users\3C\Desktop\Учёба\Программная инженерия\Практики\1 paбота\PE_1\tasks>
```

Рисунок 3 – Вывод программы третьей задачи

```
✓ Задание 4
```

**Задача:** Петя торопился в школу и неправильно написал программу, которая сначала находит квадраты двух чисел, а затем их суммирует. Исправьте его программу.

Пример ввода:

```
2 2
```

```
8
```

```
package main
import "fmt"
func main(){

  var a int
  fmt.Scan(&a) // считаем переменную 'a' с консоли
  fmt.Scan(&b) // считаем переменную 'b' с консоли

a = a * a
b = b * 2
c = a + b
fmt.Println(c)
}
```

```
package main

import "fmt"

func main() {

    // Задание 4
    var a, b, c int

    fmt.Scan(&a)
    fmt.Scan(&b)
```

```
a = a * a
b = b * b
c = a + b

fmt.Println(c)
}
```

Листинг 4 – Код решения четвертой задачи

```
PS C:\Users\3C\Desktop\Учёба\Программная инженерия\Практики\1 paбота\PE_1\tasks> go
2
2
8
PS C:\Users\3C\Desktop\Учёба\Программная инженерия\Практики\1 paбота\PE_1\tasks>
```

Рисунок 4 — Вывод программы четвертой задачи

## √ Задание 5

Задача: По данному целому числу, найдите его квадрат.

Пример ввода:

```
3
```

```
9
```

```
package main

import "fmt"

func main() {

    // Задание 5
    var num int

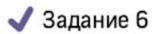
fmt.Scan(&num)
```

```
fmt.Println(num * num)
}
```

Листинг 5 – Код решения пятой задачи

```
PS C:\Users\3C\Desktop\Учёба\Программная инженерия\Практики\1 paбота\PE_1\tasks> g
3
9
PS C:\Users\3C\Desktop\Учёба\Программная инженерия\Практики\1 paбота\PE_1\tasks>
```

Рисунок 5 – Вывод программы пятой задачи



**Задача:** Дано натуральное число, выведите его последнюю цифру. На вход дается натуральное число N, не превосходящее 10000. Выведите одно целое число - ответ на задачу.

Пример ввода:

```
123
```

Соответствующий вывод:

3

```
package main

import "fmt"

func main() {

    // Задание 6
    var num int

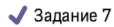
    fmt.Scan(&num)
    fmt.Println(num % 10)
```

}

### Листинг 6 – Код решения шестой задачи

```
PS C:\Users\3C\Desktop\Учёба\Программная инженерия\Практики\1 paбота\PE_1\tasks> go
123
3
PS C:\Users\3C\Desktop\Учёба\Программная инженерия\Практики\1 paбота\PE_1\tasks>
```

Рисунок 6 – Вывод программы шестой задачи



**Задача:** Дано неотрицательное целое число. Найдите число десятков (то есть вторую цифру справа). На вход дается натуральное число N, не превосходящее 10000. Выведите одно целое число - число десятков.

Пример ввода:

```
2010
```

Соответствующий вывод:

1

```
package main

import "fmt"

func main() {

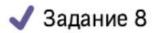
    // Задание 7
    var num int

    fmt.Scan(&num)
    fmt.Println(num % 100 / 10)
}
```

Листинг 7 – Код решения седьмой задачи

```
PS C:\Users\3C\Desktop\Учёба\Программная инженерия\Практики\1 paбота\PE_1\tasks> go 2010
1
PS C:\Users\3C\Desktop\Учёба\Программная инженерия\Практики\1 paбота\PE_1\tasks>
```

Рисунок 7 – Вывод программы седьмой задачи



**Задача:** Часовая стрелка повернулась с начала суток на d градусов. Определите, сколько сейчас целых часов h и целых минут m. На вход программе подается целое число d (0 < d < 360). Выведите на экран фразу:

```
It is ... hours ... minutes.
```

Вместо многоточий программа должна выводить значения h и m, отделяя их от слов ровно одним пробелом.

Пример ввода:

```
90
```

```
It is 3 hours 0 minutes.
```

```
package main

import "fmt"

func main() {

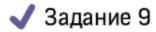
    // Задание 8
    var d uint16

    fmt.Scan(&d)
    fmt.Println("It is", d/30, "hours", d*12%360/6, "minutes")
}
```

### Листинг 8 – Код решения восьмой задачи

```
PS C:\Users\3C\Desktop\Учёба\Программная инженерия\Практики\1 paбoтa\PE_1\tasks> go
90
It is 3 hours 0 minutes
PS C:\Users\3C\Desktop\Учёба\Программная инженерия\Практики\1 paбoтa\PE_1\tasks>
```

Рисунок 8 – Вывод программы восьмой задачи



Задача: Уберите лишние комментарии так, чтобы программа вывела число 100.

Пример ввода:

Соответствующий вывод:

100

```
package main
import "fmt"
func main(){
    // a:=44
    /*
    var a2 int = 10
    */
    a2 = a2 * 10
    fmt.Println(a2)
}
```

```
package main
import "fmt"
```

```
func main() {
    // Задание 9
    // а := 44
    var a2 int = 10
    a2 = a2 * 10
    fmt.Println(a2)
}
```

Листинг 9 – Код решения девятой задачи

```
PS C:\Users\3C\Desktop\Учёба\Программная инженерия\Практики\1 paбота\PE_1\tasks> go 100
PS C:\Users\3C\Desktop\Учёба\Программная инженерия\Практики\1 paбота\PE_1\tasks>
```

Рисунок 9 – Вывод программы девятой задачи

## ✓ Задание 10

Задача: Исправьте ошибку в программе ниже:

```
package main

import "fmt"

func main(){
    var a int = 8
    const b int = 10
    a = a + b
    b = b + a
    fmt.Println(a)
}
```

Пример ввода:

```
package main
import "fmt"

func main() {
    // Задание 10
    var a int = 8
    const b int = 10
    a = a + b
    fmt.Println(a)
}
```

Листинг 10 – Код решения десятой задачи

```
PS C:\Users\3C\Desktop\Учёба\Программная инженерия\Практики\1 paбота\PE_1\tasks> go
18
PS C:\Users\3C\Desktop\Учёба\Программная инженерия\Практики\1 paбота\PE_1\tasks>
```

Рисунок 10 – Вывод программы десятой задачи

## √ Задание 11

**Задача:** Напишите программу, которая для заданных значений a и b вычисляет площадь поверхности и объем тела, образованного вращением эллипса, заданного уравнением:

$$\frac{x^2}{a^2} + \frac{y^2}{b^2} = 1,$$

вокруг оси Ox.

```
import (
    "fmt"
    "math"
)

func main() {

    // Задание 11
    const pi float64 = 3.141592
    var a, b, s, v float64

    fmt.Scan(&a, &b)

    s = pi * math.Sqrt(a) * math.Sqrt(b)
    v = (4.0 / 3) * pi * math.Sqrt(a) * b

    fmt.Println("Площадь поверхности:", s, "\nОбъем тела:", v)
}
```

Листинг 11 – Код решения одиннадцатой задачи

```
PS C:\Users\3C\Desktop\Учёба\Программная инженерия\Практики\1 paбота\PE_1\tasks> go
100 121
Площадь поверхности: 345.5751199999997
Объем тела: 5068.435093333333
```

Рисунок 11 – Вывод программы одиннадцатой задачи

#### Решение индивидуальных задач

3. **Объем пирамиды:** Задайте переменные для длины, ширины основания и высоты пирамиды. Рассчитайте и выведите объем пирамиды.

```
package main

import "fmt"

func main() {

    // Индивидуальное задание 1. Вариант 3
    var length, width, height, volume float64
```

```
fmt.Print("Введите длину, ширину основания и высоту пирамиды: ")
fmt.Scan(&length, &width, &height)

volume = (length * width * height) / 3

fmt.Print("Объём пирамиды равен:", volume)
}
```

Листинг 12 – Код решения первой индивидуальной задачи

```
PS C:\Users\3C\Desktop\Учёба\Программная инженерия\Практики\1 paбoтa\PE_1\tasks> go
Введите длину, ширину основания и высоту пирамиды: 2 2 10
Объём пирамиды равен:13.33333333333334
```

Рисунок 12 – Вывод программы первой индивидуальной задачи

 Даны стороны треугольника а, b и с найти площадь треугольника по формуле Герона

$$S = \sqrt{p(p-a)(p-b)(p-c)},$$

где p = (a + b + c) / 2.

```
package main
import (
    "fmt"
    "math"
)
func main() {
    //Индивидуальное задание 2. Вариант 3
    var a, b, c, p, s float64
    fmt.Print("Введите a, b и c:")
    fmt.Scan(&a, &b, &c)
    p = (a + b + c) / 2
    s = math.Sqrt(p * (p - a) * (p - b) * (p - c))
    fmt.Println("Площадь треугольника равна:", s)
```

```
}
```

Листинг 13 – Код решения второй индивидуальной задачи

```
PS C:\Users\3C\Desktop\Учёба\Программная инженерия\Практики\1 paботa\PE_1\tasks> go
Введите a, b и c:15 21 10
Площадь треугольника равна: 69.16646586316233
PS C:\Users\3C\Desktop\Учёба\Программная инженерия\Практики\1 paботa\PE_1\tasks>
```

Рисунок 13 – Вывод программы второй индивидуальной задачи

### Вопросы для защиты работы

1. Как объявить переменную типа int в Go?

```
var a int
a = 10

var b int = 10

c := 10
```

2. Какое значение по умолчанию присваивается переменной типа int в Go?

Ответ:0

3. Как изменить значение существующей переменной в Go?

```
var a int = 10
a = 100
```

4. Что такое множественное объявление переменных в Go?

```
var a, b, c, p, s float64
```

5. Как объявить константу в Go?

```
const a int = 10

const (
    b int = 100
    c
    e float32 = 50.0
)

const (
```

```
first int = iota
second
third
)
```

6. Можно ли изменить значение константы после ее объявления в Go?

Ответ: Нет

7. Какие арифметические операторы поддерживаются в Go?

// +-\*/% a++ a--

8. Какой оператор используется для выполнения операции остатка в Go?

Ответ: %

9. Какой результат выражения 5 / 2 в Go?

Ответ: 2

10. Как считать строку с консоли в Go?

```
var str string
fmt.Scan(&str)
```

11. Как считать целое число с консоли в Go?

```
var num int
fmt.Scan(&num)
```

12. Как обработать ошибку при считывании данных с консоли в Go?

```
_, err := fmt.Scan(&num)
if err != nil {
    // Обработка ошибки
}
```

13. Как вывести строку в консоль Go?

```
fmt.Println("I like Go!")
```

14. Как вывести значение типа int в консоль Go?

fmt.Println(num)

15. Как форматировать числа с плавающей точкой в Go?

```
fmt.Printf("%.2f\n", num) // n - число знаков после запятой
```

16. Как объявить переменную типа byte и присвоить ей значение 65? Чем отличается оператор := от оператора =?

```
var b byte = 65
```

Оператор := объявляет и инициализирует переменную с присваиванием ей значения, а оператор = только присваивает значение уже существующей переменной.

17. Какие типы данных можно использовать для представления чисел с плавающей точкой в Go?

Omeem: float32, float64

18. Как объявить и использовать несколько переменных в Go?