



Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Московский государственный технический университет
имени Н.Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н.Э. Баумана)

ФАКУЛЬТЕТ ИНФОРМАТИКА И СИСТЕМЫ УПРАВЛЕНИЯ

КАФЕДРА КОМПЬЮТЕРНЫЕ СИСТЕМЫ И СЕТИ (ИУ6)

НАПРАВЛЕНИЕ ПОДГОТОВКИ 09.03.01 Информатика и вычислительная техника

О Т Ч Е Т

по лабораторной работе № 3

Название: Основы Golang

Дисциплина: Языки интернет-программирования

Студент

ИУ6-31Б

(Группа)

(Подпись, дата)

Л. А. Круглов

(И.О. Фамилия)

Преподаватель

(Подпись, дата)

В. Д. Шульман

(И.О. Фамилия)

Москва, 2024

Цель работы: знакомство с Go, компилируемым многопоточным языком программирования

Задание:

1. Ознакомьтесь с первыми 2-я разделами курса по Golang.
2. Сделайте форк данного репозитория в GitHub, склонируйте получившуюся копию локально, создайте от мастера ветку dev и переключитесь на нее
3. Выполните задания. Ссылки на задания содержатся в README-файлах в директории projects
4. Сделайте отчёт и поместите его в директорию docs
5. Зафиксируйте изменения, сделайте коммит и отправьте полученное состояние ветки dev в удаленный репозиторий GitHub
6. Через интерфейс GitHub создайте Pull Request dev --> master
7. Защитите лабораторную работу...

Ход работы:

1. Задача №1 «Triangle»

а. Условие:

На вход подаются а и b - катеты прямоугольного треугольника. Нужно найти длину гипотенузы

Sample Input:

6 8

Sample Output:

10

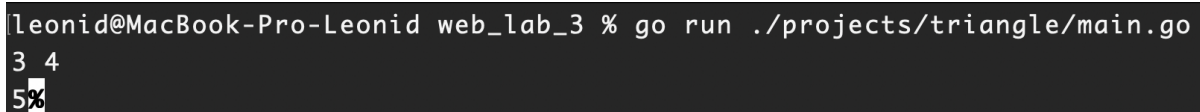
Рисунок 1 - Условие задачи №1

б. Решение:

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

func main() {
    var a, b int
    fmt.Scan(&a, &b)
    fmt.Print(math.Sqrt(float64(a * a + b * b)))
}
```

с. Тестирование:



```
leonid@MacBook-Pro-Leonid web_lab_3 % go run ./projects/triangle/main.go
3 4
5%
```

Рисунок 2 - Тестирование задачи №1

2. Задача №2 «Star»

а. Условие:

Дана строка, содержащая только английские буквы (большие и маленькие). Добавить символ '*' (звездочка) между буквами (перед первой буквой и после последней символ '*' добавлять не нужно).

Входные данные

Вводится строка ненулевой длины. Известно также, что длина строки не превышает 1000 знаков.

Выходные данные

Вывести строку, которая получится после добавления символов '*'.

Sample Input:

LI†BeoFLcSGB0FQxMHoIuDDWcqcVgkcRoAeocX0

Sample Output:

L*I†*t*B*e*o*F*L*c*S*G*B*0*F*Q*x*M*H*o*I*u*D*D*W*c*q*c*V*g*k*c*R*o*A*e*o*c*X*0

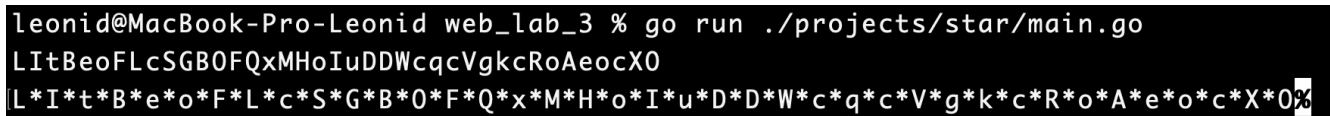
Рисунок 3 - Условие задачи №2

b. Решение:

```
package main
import "fmt"

func main() {
    var myString string
    fmt.Scan(&myString)
    for i, v := range myString {
        if i != 0 {
            fmt.Print("*")
        }
        fmt.Printf("%c", v)
    }
}
```

c. Тестирование:



```
leonid@MacBook-Pro-Leonid web_lab_3 % go run ./projects/star/main.go
LItBeoFLcSGB0FQxMHoIuDDWcqcVgkcRoAeocX0
L*I*t*B*e*o*F*L*c*S*G*B*O*F*Q*x*M*N*o*I*u*D*D*W*c*q*c*V*g*k*c*R*o*A*e*o*c*X*0%
```

Рисунок 4 - Тестирование задачи №2

3. Задача №3 «Digits»

a. Условие:

Дана строка, содержащая только арабские цифры. Найти и вывести наибольшую цифру.

Входные данные

Вводится строка ненулевой длины. Известно также, что длина строки не превышает 1000 знаков и строка содержит только арабские цифры.

Выходные данные

Выведите максимальную цифру, которая встречается во введенной строке.

Sample Input:

1112221112

Sample Output:

2

Рисунок 5 - Условие задачи №3

b. Решение:

```
package main
import "fmt"

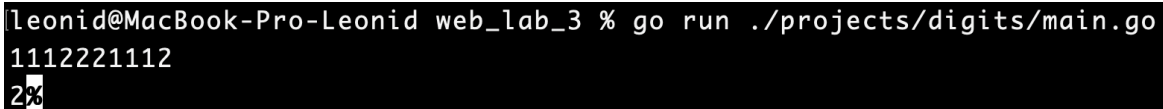
func main() {
    var myString string
    fmt.Scan(&myString)
    var mx rune
```

```

    for _, v := range myString {
        if v > mx {
            mx = v
        }
    }
    fmt.Printf("%c", mx)
}

```

с. Тестирование:



```

leonid@MacBook-Pro-Leonid web_lab_3 % go run ./projects/digits/main.go
1112221112
2%

```

Рисунок 6 - Тестирование задачи №3

4. Задача №4 «Atoi»

а. Условие:

На вход подается целое число. Необходимо возвести в квадрат каждую цифру числа и вывести получившееся число.

Например, у нас есть число 9119. Первая цифра - 9. 9 в квадрате - 81. Дальше 1. Единица в квадрате - 1. В итоге получаем 811181

Sample Input:

9119

Sample Output:

811181

Рисунок 7 - Условие задачи №4

б. Решение:

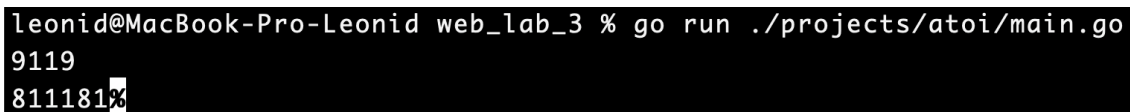
```

package main
import "fmt"

func main() {
    var n string
    fmt.Scan(&n)
    for i := 0; i < len(n); i++ {
        fmt.Print((n[i] - 48) * (n[i] - 48))
    }
}

```

с. Тестирование:



```

leonid@MacBook-Pro-Leonid web_lab_3 % go run ./projects/atoi/main.go
9119
811181%

```

Рисунок 8 - Тестирование задачи №4

5. Задачи №5 «Math»

а. Условие:

Требуется вычислить период колебаний (t) математического маятника (мы округлили некоторые значения для удобства проверки), для этого нужно найти циклическую частоту колебания пружинного маятника (w), в формуле w встречается масса которую также нужно найти, все нужные формулы приведены ниже:

Напишите три функции, каждая из которых будет выполнять конкретную формулу. Название функций обязательно должны соответствовать букве формулы: T(), W() и M(). Для того чтобы найти t - необходимо сначала найти w, и т.д. Так что используйте результат функции W() в формуле функции T() - то-есть вызывайте функцию W() в T(). Аналогично и с W(), M().

$$t = \frac{6}{w}, w = \sqrt{\frac{k}{m}}, m = p * v$$

Рисунок 9 - Условие задачи №5

б. Решение:

```
func M() float64 {  
    return p * v  
}  
func W() float64 {  
    return math.Sqrt(k / M())  
}  
func T() float64 {  
    return 6 / W()  
}
```

с. Тестирование:

Test input:

1296 6 6

Test output:

1

Запустить код

Рисунок 10 - Тестирование задачи №5

6. Изменения были загружены на удалённый репозиторий на сайте Github.
7. С помощью инструментов GitHub был выполнен Pull Request из dev в master.

Заключение: язык программирования Go позволяет решать простейшие математические задачи с использованием циклов и ветвлений.

Список источников:

- Сайт: <https://stepik.org/>
- Сайт: <https://go.dev/>