

**Московский государственный технический
университет им. Н.Э. Баумана**

Факультет «Информатика и системы управления»
Кафедра ИУ5 «Системы обработки информации и управления»

Курс «Парадигмы и конструкции языков программирования»

Отчет по лабораторной работе № 6
«Разработка телеграмм бота на Go»

Выполнил:
студент группы ИУ5-31Б
Князев А.М.

Проверил:
преподаватель каф. ИУ5
Нардид А.Н.

Москва, 2024 г.

Описание задания

1. Написать программу на языке Go, которая реализует Telegram-бота с использованием библиотеки github.com/go-telegram-bot-api/telegram-bot-api/v5. Бот должен предоставлять пользователю выбор из двух функций:
 - Найти корни квадратного уравнения: Пользователь вводит коэффициенты a , b и c квадратного уравнения, бот вычисляет и возвращает корни (действительные или комплексные).
 - Найти площадь прямоугольника: Пользователь вводит длины сторон прямоугольника, бот вычисляет и возвращает площадь.
2. Бот должен корректно обрабатывать некорректный ввод пользователя и выводить соответствующие сообщения об ошибках.

Текст программы

Файл *main.go*

```
package main

import (
    "fmt"
    "github.com/go-telegram-bot-api/telegram-bot-api/v5"
    "log"
    "math"
    "strconv"
    "strings"
)

func main() {
    bot, err := tgbotapi.NewBotAPI("7557980296:AAHWaavcV85arPbn-erWPAuEy176wm7S4Gg")
    if err != nil {
        log.Panic(err)
    }

    bot.Debug = true
    log.Printf("Authorized on account %s", bot.Self.UserName)

    u := tgbotapi.NewUpdate(0)
    u.Timeout = 60

    updates := bot.GetUpdatesChan(u)

    rootButton := tgbotapi.NewKeyboardButton("Найти корни квадратного уравнения")
    areaButton := tgbotapi.NewKeyboardButton("Найти площадь прямоугольника")
}
```

```

keyboard := tgbotapi.NewReplyKeyboard(
    tgbotapi.NewKeyboardButtonRow(rootButton),
    tgbotapi.NewKeyboardButtonRow(areaButton),
)

for update := range updates {
    if update.Message == nil {
        continue
    }

    chatID := update.Message.Chat.ID
    msg := tgbotapi.NewMessage(chatID, "")

    switch update.Message.Text {
    case "/start":
        msg.Text = "Выберите функцию:"
        msg.ReplyMarkup = keyboard
        bot.Send(msg)

    case "Найти корни квадратного уравнения":
        msg.Text = "Введите коэффициенты а, b и c через пробел (например: 1 -3 2):"
        bot.Send(msg)

    case "Найти площадь прямоугольника":
        msg.Text = "Введите длины сторон прямоугольника а и b через пробел (например:
3 4):"
        bot.Send(msg)

    default:
        switch {
        case strings.Contains(update.Message.Text, " "): // Проверяем наличие
пробелов, предполагая ввод чисел
            input := strings.Fields(update.Message.Text)
            if len(input) == 3 { // Корни квадратного уравнения
                a, _ := strconv.ParseFloat(input[0], 64)
                b, _ := strconv.ParseFloat(input[1], 64)
                c, _ := strconv.ParseFloat(input[2], 64)
                msg.Text = calculateRoots(a, b, c)
            } else if len(input) == 2 { // Площадь прямоугольника
                a, err1 := strconv.ParseFloat(input[0], 64)
                b, err2 := strconv.ParseFloat(input[1], 64)
                if err1 == nil && err2 == nil {
                    msg.Text = calculateRectangleArea(a, b)
                } else {
                    msg.Text = "Неверный формат. Пожалуйста, введите два числа через пробел."
                }
            } else {
                msg.Text = "Неверный формат ввода."
            }
        }
    }
}
default:

```

```

    msg.Text = "Пожалуйста, выберите функцию из предложенных вариантов или
введите данные в правильном формате."
}
bot.Send(msg)
}
}
}

func calculateRoots(a, b, c float64) string {
    d := b*b - 4*a*c

    if a == 0 {
        return "Уравнение не является квадратным"
    }

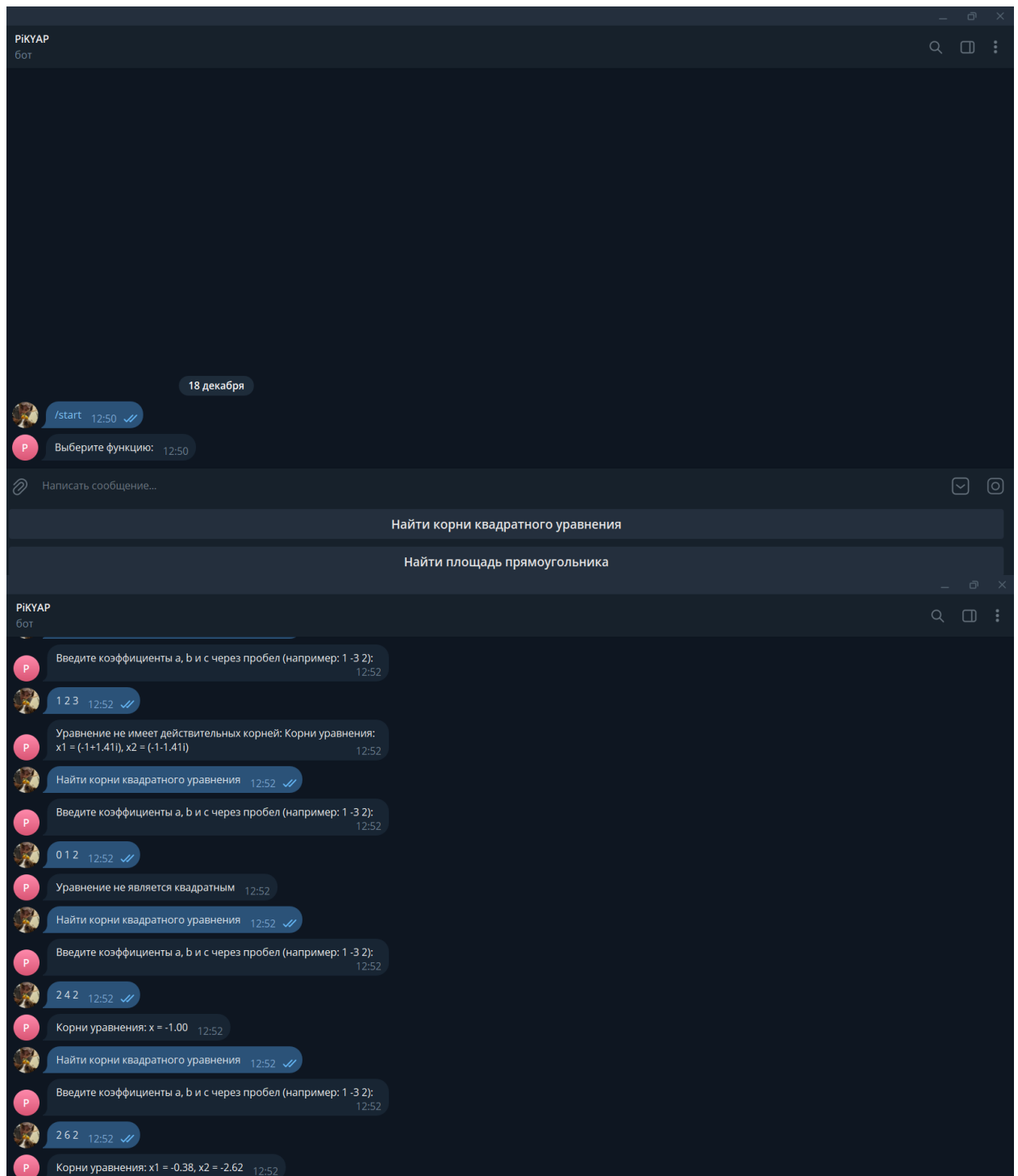
    if d > 0 {
        x1 := (-b + math.Sqrt(d)) / (2 * a)
        x2 := (-b - math.Sqrt(d)) / (2 * a)
        return fmt.Sprintf("Корни уравнения: x1 = %.2f, x2 = %.2f\n", x1, x2)

    } else if d == 0 {
        x := -b / (2 * a)
        return fmt.Sprintf("Корни уравнения: x = %.2f\n", x)
    } else {
        realPart := -b / (2 * a)
        imagPart := math.Sqrt(math.Abs(d)) / (2 * a)
        x1 := complex(realPart, imagPart)
        x2 := complex(realPart, -imagPart)
        return fmt.Sprintf("Уравнение не имеет действительных корней: Корни уравнения:
x1 = %.3v, x2 = %.3v\n", x1, x2)
    }
}

func calculateRectangleArea(a, b float64) string {
    if a <= 0 || b <= 0 {
        return "Длины сторон прямоугольника должны быть положительными числами."
    }
    area := a * b
    return fmt.Sprintf("Площадь прямоугольника: %.2f", area)
}

```

Экранные формы с примерами выполнения программы



```
lesha@lexei:~/labs/lab6$ go run main.go
2024/12/18 12:51:51 Authorized on account MoPoLoP bot
2024/12/18 12:51:51 Endpoint: getUpdates, params: map[allowed_updates:null timeout:60]
2024/12/18 12:51:59 Endpoint: getUpdates, response: {"ok":true,"result":[{"update_id":313225732,
"message":{"message_id":203,"from":{"id":1351009998,"is_bot":false,"first_name":"\u0410\u043b\u0435\u043a\u0441\u0435\u0439","username":"MoPoLoP","language_code":"ru"},"chat":{"
id":1351009998,"first_name":"\u0410\u043b\u0435\u043a\u0441\u0435\u0439","username":"MoPoLoP","type":"private"},"date":1734515519,"text":"/start","entities":[{"offset":0,"length
":6,"type":"bot_command"}]}]}]}
2024/12/18 12:51:59 Endpoint: getUpdates, params: map[allowed_updates:null offset:313225733 timeout:60]
2024/12/18 12:51:59 Endpoint: sendMessage, params: map[chat_id:1351009998 entities:null reply_markup:[{"keyboard":[{"text":"Найти корни квадратного уравнения"}], [{"text":"Найти п
лощадь прямоугольника"}], "resize_keyboard":true] text:Выберите функцию:]
2024/12/18 12:51:59 Endpoint: sendMessage, response: {"ok":true,"result":{"message_id":204,"from":{"id":7557980296,"is_bot":true,"first_name":"PiKYAP","username":"MoPoLoP_bot"},"
chat":{"id":1351009998,"first_name":"\u0410\u043b\u0435\u043a\u0441\u0435\u0439","username":"MoPoLoP","type":"private"},"date":1734515519,"text":"\u0412\u044b\u0431\u0435\u0440\u0435\u0442\u0435
\u0438\u0442\u0435 \u0444\u0443\u043d\u043a\u0446\u0438\u044e:"}}
2024/12/18 12:52:15 Endpoint: getUpdates, response: {"ok":true,"result":[{"update_id":313225733,
"message":{"message_id":205,"from":{"id":1351009998,"is_bot":false,"first_name":"\u0410\u043b\u0435\u043a\u0441\u0435\u0439","username":"MoPoLoP","language_code":"ru"},"chat":{"
id":1351009998,"first_name":"\u0410\u043b\u0435\u043a\u0441\u0435\u0439","username":"MoPoLoP","type":"private"},"date":1734515535,"text":"\u0412\u0432\u0435\u0434\u0438\u0442\u0435
\u0434\u043b\u0438\u043d\u044b \u0441\u0442\u043e\u0440\u043e\u043d \u043f\u0440\u044f\u043c\u0443\u0433\u0430 \u0430 \u0438 \u0431 \u0447\u0435\u0440\u0435\u0437 \u043f\u0440\u043e\u0431\u0435\u043b
(\u043d\u0430\u043f\u0440\u0438\u043c\u0435\u0440: 1 -3 2):}}
2024/12/18 12:52:15 Endpoint: getUpdates, params: map[allowed_updates:null offset:313225734 timeout:60]
2024/12/18 12:52:15 Endpoint: sendMessage, params: map[chat_id:1351009998 entities:null text:Введите коэффициенты а, б и с через пробел (например: 1 -3 2):]
2024/12/18 12:52:15 Endpoint: sendMessage, response: {"ok":true,"result":{"message_id":206,"from":{"id":7557980296,"is_bot":true,"first_name":"PiKYAP","username":"MoPoLoP_bot"},"
chat":{"id":1351009998,"first_name":"\u0410\u043b\u0435\u043a\u0441\u0435\u0439","username":"MoPoLoP","type":"private"},"date":1734515535,"text":"\u0412\u0432\u0435\u0434\u0438\u0442\u0435
\u0434\u043b\u0438\u043d\u044b \u0441\u0442\u043e\u0440\u043e\u043d \u043f\u0440\u044f\u043c\u0443\u0433\u0430 \u0430 \u0438 \u0431 \u0447\u0435\u0440\u0435\u0437 \u043f\u0440\u043e\u0431\u0435\u043b
(\u043d\u0430\u043f\u0440\u0438\u043c\u0435\u0440: 1 -3 2):}}
2024/12/18 12:52:18 Endpoint: getUpdates, response: {"ok":true,"result":[{"update_id":313225734,
"message":{"message_id":207,"from":{"id":1351009998,"is_bot":false,"first_name":"\u0410\u043b\u0435\u043a\u0441\u0435\u0439","username":"MoPoLoP","language_code":"ru"},"chat":{"
id":1351009998,"first_name":"\u0410\u043b\u0435\u043a\u0441\u0435\u0439","username":"MoPoLoP","type":"private"},"date":1734515538,"text":"1 2 3"}}]}
2024/12/18 12:52:18 Endpoint: getUpdates, params: map[allowed_updates:null offset:313225735 timeout:60]
2024/12/18 12:52:18 Endpoint: sendMessage, params: map[chat_id:1351009998 entities:null text:Уравнение не имеет действительных корней: Корни уравнения: x1 = (-1+1.41i), x2 = (-1-
1.41i)]
2024/12/18 12:52:19 Endpoint: sendMessage, response: {"ok":true,"result":{"message_id":208,"from":{"id":7557980296,"is_bot":true,"first_name":"PiKYAP","username":"MoPoLoP_bot"},"
chat":{"id":1351009998,"first_name":"\u0410\u043b\u0435\u043a\u0441\u0435\u0439","username":"MoPoLoP","type":"private"},"date":1734515539,"text":"\u0412\u0432\u0435\u0434\u0438\u0442\u0435
\u0434\u043b\u0438\u043d\u044b \u0441\u0442\u043e\u0440\u043e\u043d \u043f\u0440\u044f\u043c\u0443\u0433\u0430 \u0430 \u0438 \u0431 \u0447\u0435\u0440\u0435\u0437 \u043f\u0440\u043e\u0431\u0435\u043b
(\u043d\u0430\u043f\u0440\u0438\u043c\u0435\u0440: 1 -3 2):}}
2024/12/18 12:52:23 Endpoint: getUpdates, response: {"ok":true,"result":[{"update_id":313225735,
"message":{"message_id":209,"from":{"id":1351009998,"is_bot":false,"first_name":"\u0410\u043b\u0435\u043a\u0441\u0435\u0439","username":"MoPoLoP","language_code":"ru"},"chat":{"
```

