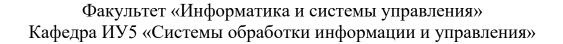
## Московский государственный технический университет им. Н.Э. Баумана



Курс «Парадигмы и конструкции языков программирования»

Отчет по лабораторной работе № 2 «Работа с массивами в Go»

Выполнил: студент группы ИУ5-31Б Князев А.М. Проверил: преподаватель каф. ИУ5 Нардид А.Н.

## Описание задания

- 1. Ввод массива:
  - Пользователь вводит элементы массива через пробел с клавиатуры.
  - Программа сохраняет их в массив.
- 2. Определение максимального и минимального элементов массива:
  - Реализовать функции *findMax* и *findMin* для поиска максимального и минимального значений.
- 3. Добавление элемента:
  - Реализовать функцию *appendElement*, которая добавляет новый элемент в конец массива.
- 4. Удаление элемента:
  - Реализовать функцию *removeElement*, которая удаляет элемент по заданному индексу.
  - Обработать некорректный ввод индекса (например, отрицательные или слишком большие значения).
- 5. Меню действий:
  - Организовать цикл, в котором пользователь может выполнять следующие действия:
    - о Добавить элемент (add)
    - о Удалить элемент (*remove*)
    - Завершить выполнение программы (exit)
- 6. Вывод результатов:

После каждого действия программа выводит обновлённый массив.

## Текст программы

Файл *lab2.go* 

```
package main

import (
    "bufio"
    "fmt"
```

```
"os"
    "strconv"
    "strings"
func findMax(arr []int) int {
    if len(arr) == 0 {
        return 0
    max := arr[0]
    for _, v := range arr {
       if v > max {
           max = v
    return max
func findMin(arr []int) int {
    if len(arr) == 0 {
        return 0
    }
    min := arr[0]
   for _, v := range arr {
       if v < min {</pre>
           min = v
        }
    }
    return min
func appendElement(arr []int, element int) []int {
   return append(arr, element)
func removeElement(arr []int, index int) []int {
    return append(arr[:index], arr[index+1:]...)
func main() {
   reader := bufio.NewReader(os.Stdin)
    fmt.Print("Введите элементы массива, разделенные пробелами: ")
    input, _ := reader.ReadString('\n')
    input = strings.TrimSpace(input)
    elements := strings.Split(input, " ")
    arr := make([]int, len(elements))
    for i, element := range elements {
```

```
num, _ := strconv.Atoi(element)
       arr[i] = num
   }
   fmt.Println("Введенный массив:", arr)
   max := findMax(arr)
   min := findMin(arr)
   fmt.Println("Максимальный элемент:", max)
   fmt.Println("Минимальный элемент:", min)
   for {
       fmt.Print("Выберите действие (add - добавить элемент, remove - удалить
элемент, exit - выход): ")
       action, _ := reader.ReadString('\n')
       action = strings.TrimSpace(action)
       switch action {
       case "add":
           fmt.Print("Введите элемент для добавления: ")
           elementStr, _ := reader.ReadString('\n')
           elementStr = strings.TrimSpace(elementStr)
           element, _ := strconv.Atoi(elementStr)
           arr = appendElement(arr, element)
           fmt.Println("Массив после добавления:", arr)
       case "remove":
           fmt.Print("Введите индекс элемента для удаления: ")
           indexStr, _ := reader.ReadString('\n')
           indexStr = strings.TrimSpace(indexStr)
           index, _ := strconv.Atoi(indexStr)
           if index < 0 || index >= len(arr) {
               fmt.Println("Неверный индекс ")
               continue
           arr = removeElement(arr, index)
           fmt.Println("Массив после удаления:", arr)
       case "exit":
           fmt.Println("Выход из программы.")
           return
       default:
           fmt.Println("Некорректное действие.")
```

## Экранные формы с примерами выполнения программы

```
lesha@Alexei:~/labs/lab2$ go run lab2.go
 Введите элементы массива, разделенные пробелами: 1 2 3 4 5
 Введенный массив: [1 2 3 4 5]
 Максимальный элемент: 5
 Минимальный элемент: 1
 Выберите действие (add - добавить элемент, remove - удалить элемент, exit - выход): add
 Введите элемент для добавления: 10
 Массив после добавления: [1 2 3 4 5 10]
 Выберите действие (add - добавить элемент, remove - удалить элемент, exit - выход): remove
 Введите индекс элемента для удаления: 4
 Массив после удаления: [1 2 3 4 10]
 Выберите действие (add - добавить элемент, remove - удалить элемент, exit - выход): exit
 Выход из программы.
○ <mark>lesha@Alexei:~/labs/lab2$</mark> go run lab2.go
 Введите элементы массива, разделенные пробелами: 1 2 3 4 5
 Введенный массив: [1 2 3 4 5]
 Максимальный элемент: 5
 Минимальный элемент: 1
 Выберите действие (add - добавить элемент, remove - удалить элемент, exit - выход): chto
 Некорректное действие.
 Выберите действие (add - добавить элемент, remove - удалить элемент, exit - выход):
lesha@Alexei:~/labs/lab2$ go run lab2.go
Введите элементы массива, разделенные пробелами: 1 2 23 4 4
Введенный массив: [1 2 23 4 4]
Максимальный элемент: 23
Минимальный элемент: 1
Выберите действие (add - добавить элемент, remove - удалить элемент, exit - выход): remove
Введите индекс элемента для удаления: -1
Неверный индекс
Выберите действие (add - добавить элемент, remove - удалить элемент, exit - выход):
```