

# Лабораторная работа 4

## Асинхронность

(10 баллов)

Выполните самостоятельно следующие задания и оформите отчет.

Требования по отчету:

Наличие титульного листа. Размер страницы должен соответствовать формату А4 (210x297), размеры полей: левое – 30 мм, правое – 10 мм, верхнее – 15 мм, нижнее – 20 мм. Шрифт Times new Roman, размер 14 pt полутонный междустрочный интервал. Выравнивание текста – по ширине, красная строка – 1,25 см, отступ слева и справа – 0 мм.

### Объекты

1. Напишите функцию `ask_password(login, password, success, failure)`, которая получает логин и пароль пользователя и проверяет их правильность. Пароль считается правильным, если в нём содержится ровно три английские гласные буквы (гласными считать буквы `a, e, i, o, u, y`) и ровно такой же набор согласных (все буквы, кроме перечисленных шести), как в логине. Порядок и количество согласных также должно совпадать. Логин и пароль приводятся к нижнему регистру и передаются.

Пример: для логина `"login"` подойдут пароли `"aaalgn"` и `"luagon"`.

`success` и `failure` – коллбэки. Коллбэком называется специальная функция, которая вызывается, когда ваше вычисление завершилось. Для программы, запустившей запрос или долгое вычисление, коллбэк – это способ сообщить, что надлежит сделать, когда вычисление завершится. Коллбэк обычно передают как аргумент функции запуска вычисления.

Если пароль правильный, функция должна вызвать коллбэк `success`, передав ему в качестве аргумента логин. А если пароль был неверный - `failure`, передав ему в качестве аргументов логин и сообщение об ошибке (в таком порядке). Сообщение об ошибке должно быть одним из трёх вариантов:

- `"Wrong number of vowels"`, если в пароле неверное число гласных;
- `"Wrong consonants"`, если в пароле набор согласных отличается от набора согласных логина;
- `"Everything is wrong"`, если оба условия нарушены.

Также напишите функцию `main(login, password)`, которая вызывает написанную функцию `ask_password` так, чтобы в случае успеха она печатала `"Привет, {логин}!"`, а в случае ошибки – `"Кто-то пытался притвориться пользователем {логин}, но в пароле допустил ошибку: {текст ошибки, большими буквами}."`

2. Следующий код эмулирует работу асинхронных функций:

```
// Асинхронные функции

function readConfig (name, callback) {
  setTimeout(() => {
    console.log('(1) config from ' + name + ' loaded')
    callback()
  }, Math.floor(Math.random() * 1000))
}

function doQuery (statement, callback) {
  setTimeout(() => {
    console.log('(2) SQL query executed: ' + statement)
    callback()
  }, Math.floor(Math.random() * 1000))
}

function httpGet (url, callback) {
  setTimeout(() => {
    console.log('(3) Page retrieved: ' + url)
    callback()
  }, Math.floor(Math.random() * 1000))
}

function readFile (path, callback) {
  setTimeout(() => {
    console.log('(4) Readme file from ' + path + ' loaded')
    callback()
  }, Math.floor(Math.random() * 1000))
}

function callback () {
  console.log('It is done!')
}

// Вызов функций

console.log('start')

readConfig('myConfig', callback)
doQuery('select * from cities', callback)
httpGet('http://google.com', callback)
readFile('README.md', callback)

console.log('end')
```

Запустите программу несколько раз и убедитесь, что функции выполняются каждый раз в разном порядке.

Перепишите вызов функций между 'start' и 'end' так, чтобы функции выполнялись в строгой последовательности: `readConfig -> doQuery -> httpGet -> readFile`. Сделайте это, используя:

- a. Коллбэки (callback)
- b. Функции-уведомители (notification)

Первоначальные асинхронные функции изменять нельзя.

3. Дана функция вида  $F(x) = f_1(x) + f_2(x) + \dots + f_n(x)$ .

Реализуйте программу, которая вычисляет значение функции  $F(x)$  и выводит его на экран, при этом все функции  $f_i(x)$  – асинхронные.

Организируйте код, в котором функции  $f_i(x)$  вычисляются последовательно от 1 до  $n$  при помощи функций-уведомителей (notification), при этом значение промежуточного результата доступно на каждом вызове и передается дальше для определения ответа.

Например,  $f_1(x) = x^2$ ,  $f_2(x) = 2x$ ,  $f_3(x) = -2$ . Тогда  $F(x) = x^2 + 2x - 2$ .

Для  $x = 3$ :

$f_1$  дает значение 9, промежуточный результат 9.

$f_2$  дает значение 6, промежуточный результат 15.

$f_3$  дает значение -2, промежуточный результат 13.

Таким образом, ответ для  $F(x)$ : 13.

Приведите примеры работы программы для:

- a.  $n = 2$
- b.  $n = 4$
- c.  $n = 6$