НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ

“КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ ІМЕНІ ІГОРЯ СІКОРСЬКОГО”

Факультет інформатики та обчислювальної техніки

Кафедра обчислювальної техніки

Лабораторна робота №1.2

з дисципліни

“Програмування мобільних систем / Розроблення клієнтських додатків для мобільних платформ”

Виконав:

студент групи ІВ-83

ЗК ІВ-8328

Чорноморець Влад

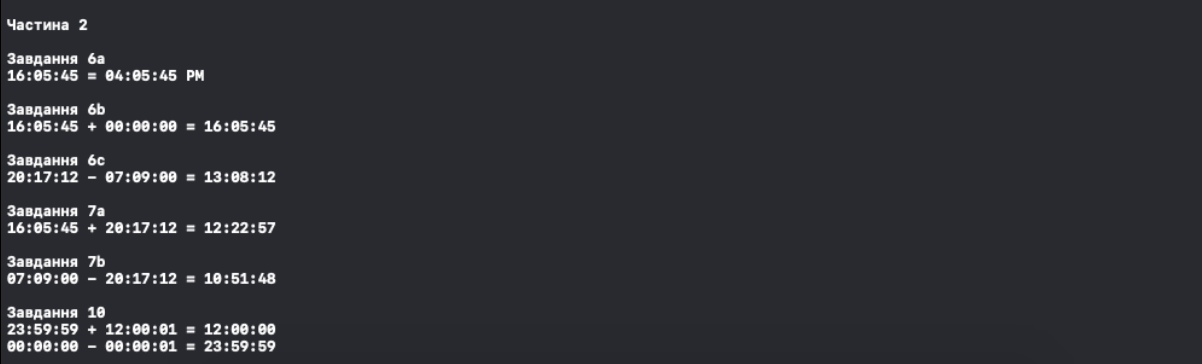
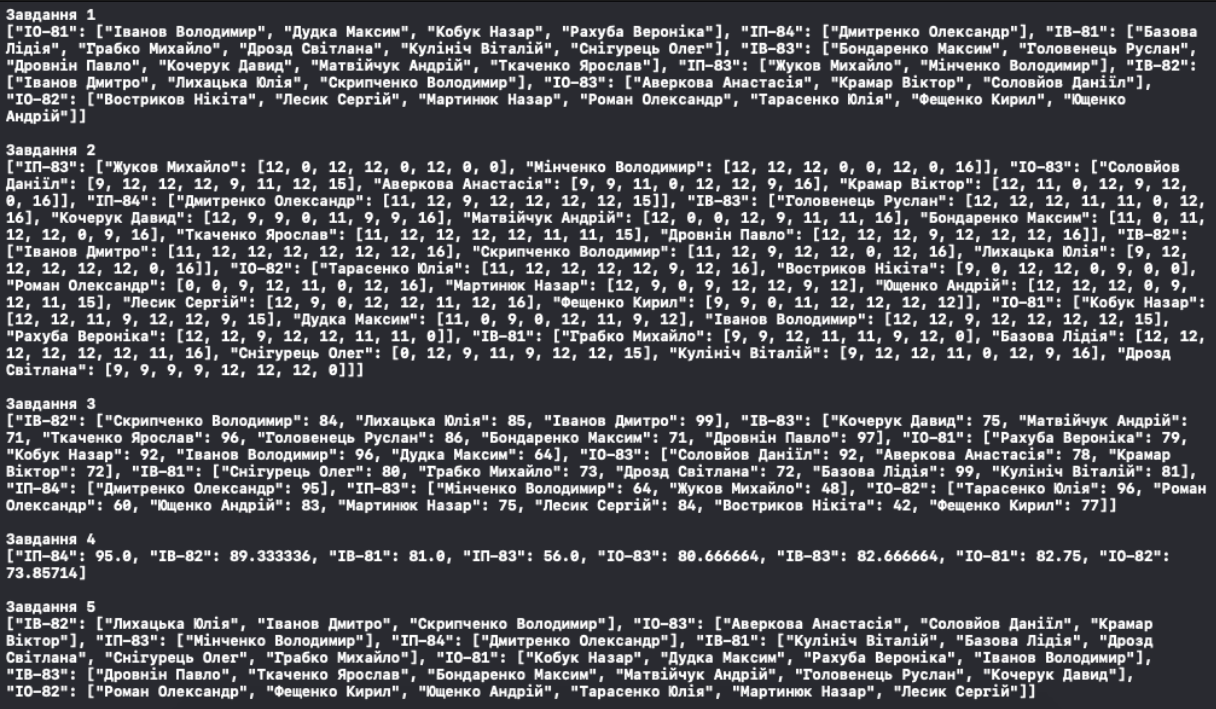
Київ 2021

**Варіант 1:**

Номер залікової книжки: 8328

8328 % 2 + 1 = 1

**Скріншот роботи додатка**



**Лістинг коду**

|  |
| --- |
| import Foundation |
|  |  |
|  | // Частина 1 |
|  |  |
|  | // Дано рядок у форматі "Student1 - Group1; Student2 - Group2; ..." |
|  |  |
|  | let studentsStr = "Дмитренко Олександр - ІП-84; Матвійчук Андрій - ІВ-83; Лесик Сергій - ІО-82; Ткаченко Ярослав - ІВ-83; Аверкова Анастасія - ІО-83; Соловйов Даніїл - ІО-83; Рахуба Вероніка - ІО-81; Кочерук Давид - ІВ-83; Лихацька Юлія - ІВ-82; Головенець Руслан - ІВ-83; Ющенко Андрій - ІО-82; Мінченко Володимир - ІП-83; Мартинюк Назар - ІО-82; Базова Лідія - ІВ-81; Снігурець Олег - ІВ-81; Роман Олександр - ІО-82; Дудка Максим - ІО-81; Кулініч Віталій - ІВ-81; Жуков Михайло - ІП-83; Грабко Михайло - ІВ-81; Іванов Володимир - ІО-81; Востриков Нікіта - ІО-82; Бондаренко Максим - ІВ-83; Скрипченко Володимир - ІВ-82; Кобук Назар - ІО-81; Дровнін Павло - ІВ-83; Тарасенко Юлія - ІО-82; Дрозд Світлана - ІВ-81; Фещенко Кирил - ІО-82; Крамар Віктор - ІО-83; Іванов Дмитро - ІВ-82" |
|  |  |
|  | // Завдання 1 |
|  | // Заповніть словник, де: |
|  | // - ключ – назва групи |
|  | // - значення – відсортований масив студентів, які відносяться до відповідної групи |
|  |  |
|  | var studentsGroups: [String: [String]] = [:] |
|  |  |
|  | // Ваш код починається тут |
|  |  |
|  | let studentsList = studentsStr.components(separatedBy: "; ") |
|  |  |
|  | for student in studentsList { |
|  |  |
|  | let nameGroup = student.components(separatedBy: "- ") |
|  | let group = nameGroup[1] |
|  | let name = String(nameGroup[0].dropLast()) |
|  |  |
|  | if studentsGroups[group] == nil{ |
|  | studentsGroups[group] = [name] |
|  | } |
|  | else{ |
|  | studentsGroups[group]!.append(name) |
|  | } |
|  | } |
|  |  |
|  | for nameOfGroup in studentsGroups.keys { |
|  | studentsGroups[nameOfGroup]!.sort{$0 < $1} |
|  | } |
|  | // Ваш код закінчується тут |
|  |  |
|  | print("Завдання 1") |
|  | print(studentsGroups) |
|  | print() |
|  |  |
|  | // Дано масив з максимально можливими оцінками |
|  |  |
|  | let points: [Int] = [12, 12, 12, 12, 12, 12, 12, 16] |
|  |  |
|  | // Завдання 2 |
|  | // Заповніть словник, де: |
|  | // - ключ – назва групи |
|  | // - значення – словник, де: |
|  | // - ключ – студент, який відносяться до відповідної групи |
|  | // - значення – масив з оцінками студента (заповніть масив випадковими значеннями, використовуючи функцію `randomValue(maxValue: Int) -> Int`) |
|  |  |
|  | func randomValue(maxValue: Int) -> Int { |
|  | switch(arc4random\_uniform(6)) { |
|  | case 1: |
|  | return Int(ceil(Float(maxValue) \* 0.7)) |
|  | case 2: |
|  | return Int(ceil(Float(maxValue) \* 0.9)) |
|  | case 3, 4, 5: |
|  | return maxValue |
|  | default: |
|  | return 0 |
|  | } |
|  | } |
|  |  |
|  | var studentPoints: [String: [String: [Int]]] = [:] |
|  |  |
|  | // Ваш код починається тут |
|  |  |
|  | for (group, students) in studentsGroups{ |
|  | var pointDict: [String:[Int]] = [:] |
|  | students.forEach { name in |
|  | let grades = points.map{randomValue(maxValue: $0)} |
|  | pointDict[name] = grades |
|  | } |
|  | studentPoints[group] = pointDict |
|  |  |
|  | } |
|  |  |
|  |  |
|  | // Ваш код закінчується тут |
|  |  |
|  | print("Завдання 2") |
|  | print(studentPoints) |
|  | print() |
|  |  |
|  | // Завдання 3 |
|  | // Заповніть словник, де: |
|  | // - ключ – назва групи |
|  | // - значення – словник, де: |
|  | // - ключ – студент, який відносяться до відповідної групи |
|  | // - значення – сума оцінок студента |
|  |  |
|  | var sumPoints: [String: [String: Int]] = [:] |
|  |  |
|  | // Ваш код починається тут |
|  |  |
|  | for (group, students) in studentPoints{ |
|  | var sumsOfGrades: [String : Int] = [:] |
|  | students.forEach{name,points in |
|  | let sum = points.reduce(0){$0 + $1} |
|  | sumsOfGrades[name]=sum |
|  | } |
|  | sumPoints[group] = sumsOfGrades |
|  |  |
|  | } |
|  |  |
|  | // Ваш код закінчується тут |
|  |  |
|  | print("Завдання 3") |
|  | print(sumPoints) |
|  | print() |
|  |  |
|  | // Завдання 4 |
|  | // Заповніть словник, де: |
|  | // - ключ – назва групи |
|  | // - значення – середня оцінка всіх студентів групи |
|  |  |
|  | var groupAvg: [String: Float] = [:] |
|  |  |
|  | // Ваш код починається тут |
|  |  |
|  | for (group,students) in sumPoints{ |
|  | var sum: Float = 0.0; |
|  | students.forEach {\_,points in |
|  | sum += Float(points) |
|  | } |
|  | groupAvg[group] = sum/Float(students.count) |
|  | } |
|  |  |
|  |  |
|  | // Ваш код закінчується тут |
|  |  |
|  | print("Завдання 4") |
|  | print(groupAvg) |
|  | print() |
|  |  |
|  | // Завдання 5 |
|  | // Заповніть словник, де: |
|  | // - ключ – назва групи |
|  | // - значення – масив студентів, які мають >= 60 балів |
|  |  |
|  | var passedPerGroup: [String: [String]] = [:] |
|  |  |
|  | // Ваш код починається тут |
|  |  |
|  | for (group,students) in sumPoints{ |
|  | let goodStudents = students.filter{$0.1 >= 60} |
|  | passedPerGroup[group] = Array(goodStudents.keys) |
|  | } |
|  |  |
|  | // Ваш код закінчується тут |
|  |  |
|  | print("Завдання 5") |
|  | print(passedPerGroup) |
|  |  |
|  | // Приклад виведення. Ваш результат буде відрізнятися від прикладу через використання функції random для заповнення масиву оцінок та через інші вхідні дані. |
|  | // |
|  | //Завдання 1 |
|  | //["ІВ-73": ["Гончар Юрій", "Давиденко Костянтин", "Капінус Артем", "Науменко Павло", "Чередніченко Владислав"], "ІВ-72": ["Бортнік Василь", "Киба Олег", "Овчарова Юстіна", "Тимко Андрій"], "ІВ-71": ["Андрющенко Данило", "Гуменюк Олександр", "Корнійчук Ольга", "Музика Олександр", "Трудов Антон", "Феофанов Іван"]] |
|  | // |
|  | //Завдання 2 |
|  | //["ІВ-73": ["Давиденко Костянтин": [5, 8, 9, 12, 11, 12, 0, 0, 14], "Капінус Артем": [5, 8, 12, 12, 0, 12, 12, 12, 11], "Науменко Павло": [4, 8, 0, 12, 12, 11, 12, 12, 15], "Чередніченко Владислав": [5, 8, 12, 12, 11, 12, 12, 12, 15], "Гончар Юрій": [5, 6, 0, 12, 0, 11, 12, 11, 14]], "ІВ-71": ["Корнійчук Ольга": [0, 0, 12, 9, 11, 11, 9, 12, 15], "Музика Олександр": [5, 8, 12, 0, 11, 12, 0, 9, 15], "Гуменюк Олександр": [5, 8, 12, 9, 12, 12, 11, 12, 15], "Трудов Антон": [5, 0, 0, 11, 11, 0, 12, 12, 15], "Андрющенко Данило": [5, 6, 0, 12, 12, 12, 0, 9, 15], "Феофанов Іван": [5, 8, 12, 9, 12, 9, 11, 12, 14]], "ІВ-72": ["Киба Олег": [5, 8, 12, 12, 11, 12, 0, 0, 11], "Овчарова Юстіна": [5, 8, 12, 0, 11, 12, 12, 12, 15], "Бортнік Василь": [4, 8, 12, 12, 0, 12, 9, 12, 15], "Тимко Андрій": [0, 8, 11, 0, 12, 12, 9, 12, 15]]] |
|  | // |
|  | //Завдання 3 |
|  | //["ІВ-72": ["Бортнік Василь": 84, "Тимко Андрій": 79, "Овчарова Юстіна": 87, "Киба Олег": 71], "ІВ-73": ["Капінус Артем": 84, "Науменко Павло": 86, "Чередніченко Владислав": 99, "Гончар Юрій": 71, "Давиденко Костянтин": 71], "ІВ-71": ["Корнійчук Ольга": 79, "Трудов Антон": 66, "Андрющенко Данило": 71, "Гуменюк Олександр": 96, "Феофанов Іван": 92, "Музика Олександр": 72]] |
|  | // |
|  | //Завдання 4 |
|  | //["ІВ-71": 79.333336, "ІВ-72": 80.25, "ІВ-73": 82.2] |
|  | // |
|  | //Завдання 5 |
|  | //["ІВ-72": ["Бортнік Василь", "Киба Олег", "Овчарова Юстіна", "Тимко Андрій"], "ІВ-73": ["Давиденко Костянтин", "Капінус Артем", "Чередніченко Владислав", "Гончар Юрій", "Науменко Павло"], "ІВ-71": ["Музика Олександр", "Трудов Антон", "Гуменюк Олександр", "Феофанов Іван", "Андрющенко Данило", "Корнійчук Ольга"]] |
|  |  |
|  | //Код до частини 2/ Варiант 1 (Залiкова книжка 8328 % 2 + 1 = 1) |
|  | //Завдання 3 |
|  | class TimeVC { |
|  | //Завдання 4 |
|  | var hours: UInt |
|  | var minutes: UInt |
|  | var seconds: UInt |
|  | //Завдання 5а |
|  | init() { |
|  | self.hours = 0 |
|  | self.minutes = 0 |
|  | self.seconds = 0 |
|  | } |
|  | //Завдання 5b |
|  | init?(hours: UInt, minutes: UInt, seconds: UInt) { |
|  | if(hours < 0 || hours > 23 || minutes < 0 || minutes > 59 || seconds < 0 || seconds > 59) { |
|  | return nil |
|  | } |
|  | self.hours = hours |
|  | self.minutes = minutes |
|  | self.seconds = seconds |
|  | } |
|  | //Завдання 5с |
|  | init(date: Date) { |
|  | self.hours = UInt(Calendar.current.component(.hour, from: date)) |
|  | self.minutes = UInt(Calendar.current.component(.minute, from: date)) |
|  | self.seconds = UInt(Calendar.current.component(.second, from: date)) |
|  | } |
|  |  |
|  | func get24Time() -> String { |
|  | return "\(hours < 10 ? "0" : "")\(hours):\(minutes < 10 ? "0" : "")\(minutes):\(seconds < 10 ? "0" : "")\(seconds)" |
|  | } |
|  | } |
|  | //Завдання 6 |
|  | extension TimeVC { |
|  | //Завдання 6а |
|  | func get12Time6a() -> String { |
|  | if(hours >= 12) { |
|  | let hours12 = String(hours == 12 ? 12 : (hours - 12)) |
|  | return "\(hours12.count == 1 ? "0" : "")\(hours12):\(minutes < 10 ? "0" : "")\(minutes):\(seconds < 10 ? "0" : "")\(seconds) PM" |
|  | } else { |
|  | return "\(hours < 10 ? "0" : "")\(hours):\(minutes < 10 ? "0" : "")\(minutes):\(seconds < 10 ? "0" : "")\(seconds) AM" |
|  | } |
|  | } |
|  | //Завдання 6b |
|  | func getSum6b(secondObject: TimeVC) -> TimeVC{ |
|  | var secondsSum = self.seconds + secondObject.seconds |
|  | var minutesSum = (secondsSum >= 60 ? 1 : 0) + self.minutes + secondObject.minutes |
|  | let hoursSum = ((minutesSum >= 60 ? 1 : 0) + self.hours + secondObject.hours) % 24 |
|  | minutesSum = minutesSum % 60 |
|  | secondsSum = secondsSum % 60 |
|  |  |
|  | let finalObject = TimeVC(hours: hoursSum, minutes: minutesSum, seconds: secondsSum) |
|  | return finalObject! |
|  | } |
|  | //Завдання 6с |
|  | func getDifference6c(secondObject: TimeVC) -> TimeVC{ |
|  | var secondsDifference = Int(self.seconds) - Int(secondObject.seconds) |
|  | var minutesDifference = Int(self.minutes) - Int(secondObject.minutes) - (secondsDifference < 0 ? 1 : 0) |
|  | var hoursDifference = Int(self.hours) - Int(secondObject.hours) - (minutesDifference < 0 ? 1 : 0) |
|  | secondsDifference = (secondsDifference < 0 ? (60 + secondsDifference) : secondsDifference) |
|  | minutesDifference = (minutesDifference < 0 ? (60 + minutesDifference) : minutesDifference) |
|  | hoursDifference = (hoursDifference < 0 ? (24 + hoursDifference) : hoursDifference) |
|  |  |
|  | let finalObject = TimeVC(hours: UInt(hoursDifference), minutes: UInt(minutesDifference), seconds: UInt(secondsDifference)) |
|  | return finalObject! |
|  | } |
|  | } |
|  | //Завдання 7 |
|  | extension TimeVC{ |
|  | //Завдання 7а |
|  | static func getSum7a(firstObject: TimeVC, secondObject: TimeVC) -> TimeVC{ |
|  | return firstObject.getSum6b(secondObject: secondObject) |
|  | } |
|  | //Завдання 7b |
|  | static func getDifference7b(firstObject: TimeVC, secondObject: TimeVC) -> TimeVC{ |
|  | return firstObject.getDifference6c(secondObject: secondObject) |
|  | } |
|  | } |
|  | print() |
|  | print("Частина 2") |
|  | var time1 = TimeVC(hours: 16, minutes:5, seconds: 45) |
|  | var time2 = TimeVC(hours: 20, minutes: 17, seconds: 12) |
|  | var time3 = TimeVC(hours: 23, minutes: 59, seconds: 59) |
|  | var time4 = TimeVC(hours: 12, minutes: 0, seconds: 1) |
|  | var time5 = TimeVC(hours: 00, minutes: 00, seconds: 01) |
|  | var timeEmpty = TimeVC() |
|  | var timeDate = TimeVC(date: Date()) |
|  |  |
|  | print("\nЗавдання 6a") |
|  | print(time1!.get24Time() + " = " + time1!.get12Time6a()) |
|  | print("\nЗавдання 6b") |
|  | print(time1!.get24Time() + " + " + timeEmpty.get24Time() + " = " + time1!.getSum6b(secondObject: timeEmpty).get24Time()) |
|  | print("\nЗавдання 6c") |
|  | print(time2!.get24Time() + " - " + timeDate.get24Time() + " = " + time2!.getDifference6c(secondObject: timeDate).get24Time()) |
|  | print("\nЗавдання 7a") |
|  | print(time1!.get24Time() + " + " + time2!.get24Time() + " = " + TimeVC.getSum7a(firstObject: time1!, secondObject: time2!).get24Time()) |
|  | print("\nЗавдання 7b") |
|  | print(timeDate.get24Time() + " - " + time2!.get24Time() + " = " + TimeVC.getDifference7b(firstObject: timeDate, secondObject: time2!).get24Time()) |
|  |  |
|  | print("\nЗавдання 10") |
|  | print(time3!.get24Time() + " + " + time4!.get24Time() + " = " + time3!.getSum6b(secondObject: time4!).get24Time()) |
|  | print(timeEmpty.get24Time() + " - " + time5!.get24Time() + " = " + timeEmpty.getDifference6c(secondObject: time5!).get24Time()) |

**Висновок:**

Під час виконання лаб.роботи 1.2 було вивчено основи Swift. Було використано різні операції та методи взаємодії зі словниками, створено клас та його методи за варіантом.