

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 ІО-82
ЗК ІО-8220
Самутін Олексій

Варіант № 1

Скопійований з консолі результат виконання частини 1

C:\Users\Alex\Desktop\University\mobile
development\firstProject\Components\Playground> node .\Playground.js

Завдання 1

```
{
  'ІВ-81': [
    'Базова Лідія',
    'Грабко Михайло',
    'Дрозд Світлана',
    'Кулініч Віталій',
    'Снігурець Олег'
  ],
  'ІВ-82': [ 'Іванов Дмитро', 'Лихацька Юлія', 'Скрипченко Володимир' ],
  'ІВ-83': [
    'Бондаренко Максим',
    'Головенець Руслан',
    'Дровнін Павло',
    'Кочерук Давид',
    'Матвійчук Андрій',
    'Ткаченко Ярослав'
  ],
  'ІО-81': [
    'Іванов Володимир',
    'Дудка Максим',
    'Кобук Назар',
    'Рахуба Вероніка'
  ],
  'ІО-82': [
    'Востриков Нікіта',
    'Лесик Сергій',
    'Мартинюк Назар',
    'Роман Олександр',
    'Тарасенко Юлія',
    'Фещенко Кирил',
    'Ющенко Андрій'
  ],
}
```

```
'Ю-83': [ 'Аверкова Анастасія', 'Крамар Віктор', 'Соловійов Даніїл' ],  
'ПІ-83': [ 'Жуков Михайло', 'Мінченко Володимир' ],  
'ПІ-84': [ 'Дмитренко Олександр' ]  
}
```

Завдання 2

```
{  
  'ІВ-81': {  
    'Базова Лідія': [  
      0, 0, 11, 0,  
      11, 0, 12, 12  
    ],  
    'Грабко Михайло': [  
      12, 9, 0, 0,  
      11, 0, 0, 0  
    ],  
    'Дрозд Світлана': [  
      11, 0, 9, 0,  
      0, 11, 0, 16  
    ],  
    'Кулініч Віталій': [  
      0, 9, 0, 11,  
      0, 0, 9, 0  
    ],  
    'Снігурець Олег': [  
      0, 0, 9, 0,  
      9, 0, 11, 0  
    ]  
  },  
  'ІВ-82': {  
    'Іванов Дмитро': [  
      11, 9, 0, 0,  
      12, 11, 0, 0  
    ],  
    'Лихацька Юлія': [  
      0, 9, 0, 0,  
      0, 9, 0, 0  
    ],  
  }  
}
```

'Скрипченко Володимир': [
0, 11, 0, 0,
9, 0, 9, 15
]
,
'ІВ-83': {
'Бондаренко Максим': [
12, 0, 0, 11,
0, 0, 9, 0
],
'Головенець Руслан': [
0, 0, 11, 0,
0, 9, 0, 15
],
'Дровнін Павло': [
11, 9, 0, 11,
0, 9, 0, 0
],
'Кочерук Давид': [
12, 9, 9, 9,
0, 11, 0, 0
],
'Матвійчук Андрій': [
0, 0, 0, 0,
0, 0, 12, 12
],
'Ткаченко Ярослав': [
0, 0, 0, 0,
9, 9, 9, 0
]
,
'ІО-81': {
'Іванов Володимир': [
9, 0, 11, 0,
11, 0, 0, 0
],
'Дудка Максим': [

```
    11, 0, 0, 0,
    0, 12, 0, 0
],
'Кобук Назар': [
    0, 0, 0, 0,
    11, 9, 11, 0
],
'Рахуба Вероніка': [
    9, 9, 11, 0,
    0, 11, 0, 16
]
},
'Ю-82': {
    'Востриков Нікіта': [
        0, 9, 0, 0,
        0, 11, 9, 0
    ],
    'Лесик Сергій': [
        12, 9, 0, 11,
        0, 0, 0, 0
    ],
    'Мартинюк Назар': [
        11, 12, 11, 0,
        11, 0, 0, 0
    ],
    'Роман Олександр': [
        11, 0, 0, 0,
        12, 11, 0, 0
    ],
    'Тарасенко Юлія': [
        0, 0, 0, 12,
        0, 0, 0, 0
    ],
    'Фещенко Кирил': [
        0, 11, 0, 0,
        0, 0, 0, 12
    ],
}
```

```

'Ющенко Андрій': [
    0, 0, 9, 9,
    11, 0, 0, 15
]
},
'Ю-83': {
    'Аверкова Анастасія': [
        9, 9, 9, 0,
        0, 0, 9, 12
    ],
    'Крамар Віктор': [
        12, 12, 11, 11,
        12, 0, 0, 12
    ],
    'Соловійов Данііл': [
        0, 0, 0, 11,
        11, 0, 12, 0
    ]
},
'П-83': {
    'Жуков Михайло': [
        9, 9, 0, 11,
        0, 0, 0, 16
    ],
    'Мінченко Володимир': [
        0, 0, 0, 12,
        0, 0, 11, 15
    ]
},
'П-84': { 'Дмитренко Олександр': [
    0, 0, 0, 0,
    0, 0, 0, 0
] }
}

```

Завдання 3

```
{
```

'В-81': {

'Базова Лідія': 46,

'Грабко Михайло': 32,

'Дрозд Світлана': 47,

'Кулініч Віталій': 29,

'Снігурець Олег': 29

},

'В-82': {

'Іванов Дмитро': 43,

'Лихацька Юлія': 18,

'Скрипченко Володимир': 44

},

'В-83': {

'Бондаренко Максим': 32,

'Головенець Руслан': 35,

'Дровнін Павло': 40,

'Кочерук Давид': 50,

'Матвійчук Андрій': 24,

'Ткаченко Ярослав': 27

},

'Ю-81': {

'Іванов Володимир': 31,

'Дудка Максим': 23,

'Кобук Назар': 31,

'Рахуба Вероніка': 56

},

'Ю-82': {

'Востриков Нікіта': 29,

'Лесик Сергій': 32,

'Мартинюк Назар': 45,

'Роман Олександр': 34,

'Тарасенко Юлія': 12,

'Фещенко Кирил': 23,

'Ющенко Андрій': 44

},

'Ю-83': {

'Аверкова Анастасія': 48,

```
'Крамар Віктор': 70,  
'Соловійов Даніїл': 34  
,  
'ПІ-83': { 'Жуков Михайло': 45, 'Мінченко Володимир': 38 },  
'ПІ-84': { 'Дмитренко Олександр': 0 }  
}
```

Завдання 4

```
{  
'ІВ-81': 36.6,  
'ІВ-82': 35,  
'ІВ-83': 34.6666666666666664,  
'ІО-81': 35.25,  
'ІО-82': 31.285714285714285,  
'ІО-83': 50.6666666666666664,  
'ПІ-83': 41.5,  
'ПІ-84': 0  
}
```

Завдання 5

```
{  
'ІВ-81': [],  
'ІВ-82': [],  
'ІВ-83': [],  
'ІО-81': [],  
'ІО-82': [],  
'ІО-83': [ 'Крамар Віктор' ],  
'ПІ-83': [],  
'ПІ-84': []  
}
```

Скопійований з консолі результат виконання частини 2

```
PS C:\Users\Alex\Desktop\University\mobile  
development\firstProject\Components\Playground> node .\TimesOS.js  
12:00:00 PM  
TimeOS { hours: 11, minutes: 59, seconds: 59 }  
TimeOS { hours: 0, minutes: 0, seconds: 0 }
```


Лістинг коду

Playground.js

```
let studentsStr =
  "Дмитренко Олександр - ІП-84; Матвійчук Андрій - ІВ-
83; Лесик Сергій - ІО-82; Ткаченко Ярослав - ІВ-
83; Аверкова Анастасія - ІО-83; Соловйов Даниїл - 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 відпов
ідної групи
let studentsGroups = {};
// Ваш код починається тут
const arrayStudentGroup = studentsStr
  .split(";")
  .map((student) => {
    return student
      .trim()
      .split(/^(^-[+])/)
      .slice(1);
  })
  .map((student) => {
    return [student[0].trim(), student[1].replace("- ", "")];
  });
const groups = [...new Set(arrayStudentGroup.map((elem) => elem[1])).sort
()];
groups.forEach((group) => {
  studentsGroups[group] = [];
});
arrayStudentGroup.forEach((studentAndGroup) => {
  const [student, group] = studentAndGroup;
  studentsGroups[group].push(student);
});
for (let group in studentsGroups) {
  studentsGroups[group].sort();
}
// Ваш код закінчується тут

console.log("Завдання 1")
console.log(studentsGroups)
```

```

// Дано масив з максимально можливими оцінками

let points = [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`)

function randomValue(maxValue) {
  switch (Math.round(Math.random() * 5)) {
    case 1:
      return Math.ceil(maxValue * 0.7);
      break;
    case 2:
      return Math.ceil(maxValue * 0.9);
      break;
    case (3, 4, 5):
      return maxValue;
      break;
    default:
      return 0;
  }
}

let studentPoints = {};

// Ваш код починається тут
for (let group in studentsGroups) {
  studentPoints[group] = {};
  const arrGroupWithPoint = studentsGroups[group]
    .map((student) => {
      let studentPoints = [];
      points.forEach((point) => {
        studentPoints.push(randomValue(point));
      });
      return { [student]: studentPoints };
    })
    .forEach((studentWithPoints) => {
      const [studentName, points] = [
        Object.keys(studentWithPoints)[0],
        Object.values(studentWithPoints)[0],
      ];
      studentPoints[group][studentName] = points;
    });
}

// Ваш код закінчується тут

```

```

console.log("Завдання 2");
console.log(studentPoints);
console.log();

// Завдання 3
// Заповніть словник, де:
// - ключ - назва групи
// - значення - словник, де:
//   - ключ - студент, який відносяться до відповідної групи
//   - значення - сума оцінок студента

let sumPoints = {};

// Ваш код починається тут
for (let group in studentPoints) {
  sumPoints[group] = {};
  for (let student in studentPoints[group]) {
    sumPoints[group][student] = studentPoints[group][student].reduce(
      (a, b) => a + b,
      0
    );
  }
}

// Ваш код закінчується тут
console.log("Завдання 3");
console.log(sumPoints);
console.log();

// Завдання 4
// Заповніть словник, де:
// - ключ - назва групи
// - значення - середня оцінка всіх студентів групи

let groupAvg = {};
// Ваш код починається тут
for (let group in sumPoints) {
  const numberOfStudentsInGroup = Object.values(sumPoints[group]).length
  ;
  groupAvg[group] =
    Object.values(sumPoints[group]).reduce((a, b) => a + b, 0) /
    numberOfStudentsInGroup;
}
// Ваш код закінчується тут

console.log("Завдання 4");
console.log(groupAvg);
console.log();

// Завдання 5
// Заповніть словник, де:
// - ключ - назва групи

```

```
// - значення – масив студентів, які мають >= 60 балів

let passedPerGroup = {}

// Ваш код починається тут
for (let group in sumPoints){
  passedPerGroup[group] = [];
  for (let student in sumPoints[group]){
    if (sumPoints[group][student] >= 60){
      passedPerGroup[group].push(student)
    }
  }
}

// Ваш код закінчується тут

console.log("Завдання 5")
console.log(passedPerGroup)
```

TimeOS.js

```
class TimeOS {
  constructor() {
    if (arguments.length === 0) {
      this.hours = 0;
      this.minutes = 0;
      this.seconds = 0;
    } else if (arguments.length === 1) {
      if (
        arguments[0] instanceof Date &&
        typeof arguments[0].getMonth === "function"
      ) {
        this.hours = arguments[0].getHours();
        this.minutes = arguments[0].getMinutes();
        this.seconds = arguments[0].getSeconds();
      } else {
        throw new Error("Invalid date object");
      }
    } else if (arguments.length === 3) {
      const [hours, minutes, seconds] = arguments;
      if (
        typeof hours !== "number" ||
        typeof minutes !== "number" ||
        typeof seconds !== "number"
      ) {
        throw new Error("Parameters should be numbers");
      }
      if (hours >= 0 && hours <= 23) {
        this.hours = hours;
      } else {
```

```

        throw new Error(
            "Invalid hours parameter, it should be greater than 0 and less
than 23"
        );
    }
    if (minutes >= 0 && minutes <= 60) {
        this.minutes = minutes;
    } else {
        throw new Error(
            "Invalid minutes parameter, it should be greater than 0 and le
ss than 60"
        );
    }
    if (seconds >= 0 && seconds <= 60) {
        this.seconds = seconds;
    } else {
        throw new Error(
            "Invalid seconds parameter, it should be greater than 0 and le
ss than 60"
        );
    }
}

formattedDate() {
    if (this.hours === 0) {
        return `12:${this.minutes < 10 ? "0" + this.minutes : this.minutes
}:${
        this.seconds < 10 ? "0" + this.seconds : this.seconds
    } AM`;
    }
    if (this.hours === 12) {
        return `12:${this.minutes < 10 ? "0" + this.minutes : this.minutes
}:${
        this.seconds < 10 ? "0" + this.seconds : this.seconds
    } PM`;
    }
    return `${
        this.hours <= 12
        ? this.hours < 10
        ? "0" + this.hours
        : this.hours
        : this.hours - 12
    }:${this.minutes < 10 ? "0" + this.minutes : this.minutes}:${
        this.seconds < 10 ? "0" + this.seconds : this.seconds
    } ${this.hours <= 12 ? "AM" : "PM"}`;
}

static sumOfDate(dateObj1, dateObj2) {
    const [hours, minutes, seconds] = [
        dateObj1.hours,

```

```

        dateObj1.minutes,
        dateObj1.seconds,
    ];
    const [hours2, minutes2, seconds2] = [
        dateObj2.hours,
        dateObj2.minutes,
        dateObj2.seconds
    ]
    let resultHours = 0,
        resultMinutes = 0,
        resultSeconds = 0;
    resultHours = hours + hours2;
    resultMinutes = minutes + minutes2;
    resultSeconds = seconds + seconds2;
    if (resultSeconds > 59) {
        resultSeconds -= 60;
        resultMinutes += 1;
    }
    if (resultMinutes > 59){
        resultMinutes -= 60;
        resultHours += 1;
    }
    if (resultHours > 23){
        resultHours -= 24
    }
    return new TimeOS(resultHours,resultMinutes,resultSeconds)
}

```

```

static differenceBetweenDate(dateObj1,dateObj2) {
    const [hours, minutes, seconds] = [
        dateObj1.hours,
        dateObj1.minutes,
        dateObj1.seconds,
    ];
    const [hours2, minutes2, seconds2] = [
        dateObj2.hours,
        dateObj2.minutes,
        dateObj2.seconds,
    ];
    let resultHours = 0,
        resultMinutes = 0,
        resultSeconds = 0;
    resultHours = hours - hours2;
    resultMinutes = minutes - minutes2;
    resultSeconds = seconds - seconds2;
    if (resultSeconds < 0){
        resultSeconds += 60;
        resultMinutes -= 1;
    }
    if (resultMinutes < 0){
        resultMinutes += 60;
    }
}

```

```

        resultHours -= 0;
    }
    if (resultHours < 0){
        resultHours += 23;
    }
    return new TimeOS(resultHours, resultMinutes, resultSeconds);
}
}

// for (let i = 0; i < 24; i++) {
//     const time1 = new TimeOS(i, 0, 0);
//     console.log(time1.formattedDate());
// }
const time3 = new TimeOS(23, 59, 59);
const time4 = new TimeOS(12, 0, 0);
console.log(time4.formattedDate());
console.log(TimeOS.sumOfDate(time3,time4));
const time5 = new TimeOS(0, 0, 0);
const time6 = new TimeOS(0, 0, 0);
console.log(TimeOS.differenceBetweenDate(time5,time6));
const time7 = new TimeOS(new Date("December 17, 1995 00:00:00"));
const time8 = new TimeOS(new Date("December 17, 1995 00:00:01"));
const time9 = new TimeOS();

```

Висновки

Під час виконання даної лабораторної роботи я навчився основам мови JavaScript.