**Лабораторная работа 3**

**Классы в языке Swift**

Разработать программу, выполняющую следующие действия:

**Вариант 1.**

− Ввод с клавиатуры данных о студентах в массив объектов класса Student.

− Вывод списка всех студентов с результатами сдачи сессии в алфавитном порядке.

− Определение количества двоечников в массиве.

− Вывод списка отличников (если таких студентов нет, вывести соответствующее сообщение) в каждой группе.

Класс Student должен содержать поля: фамилия, номер группы, успеваемость (массив), сабскрипт для доступа к элементам массива оценок и все необходимые для решения задачи свойства и методы.

**Вариант 2.**

− Ввод с клавиатуры данных о студентах в массив объектов класса Student.

− Вывод списка всех студентов заданной группы с результатами сдачи сессии в алфавитном порядке.

− Определение количества отличников в каждой группе.

− Вывод списка всех двоечников (если таких студентов нет, вывести соответствующее сообщение).

Класс Student должен содержать закрытые поля: фамилия, номер группы, успеваемость (массив), сабскрипт для доступа к элементам массива оценок и все необходимые для решения задачи свойства и методы.

**Вариант 3.**

− Ввод с клавиатуры данных о студентах в массив объектов класса Student.

− Вывод списка всех студентов с указанием среднего балла каждого студента в алфавитном порядке.

− Определение количества студентов, получивших больше двух оценок 10 в массиве.

− Вывод списка двоечников в заданной группе (если таких студентов нет, вывести соответствующее сообщение).

Класс Student должен содержать закрытые поля: фамилия, номер группы, успеваемость (массив), сабскрипт для доступа к элементам массива оценок и все необходимые для решения задачи свойства и методы.

**Вариант 4.**

− Ввод с клавиатуры данных о спортсменах в массив объектов класса Sportsman.

− Вывод списка всех спортсменов с указанием результатов соревнований каждого спортсмена в алфавитном порядке.

− Определение среднего результата соревнований для каждой возрастной группы.

− Вывод списка спортсменов, результат которых был меньше

заданного не менее одного раза за соревнования (если таких спортсменов нет, вывести соответствующее сообщение).

Класс Sportsman должен содержать закрытые поля: фамилия, год рождения, результаты соревнований (массив), сабскрипт для доступа к элементам массива результатов и все необходимые для решения задачи свойства и методы.

**Вариант 5.**

− Ввод с клавиатуры данных о спортсменах в массив объектов класса Sportsman.

− Вывод списка всех спортсменов с указанием среднего результата соревнований каждого спортсмена в порядке возрастания года рождения.

− Определение количества спортсменов, средний результат которых превысил заданное значение.

− Вывод списка спортсменов, минимальный результат которых был в заданном диапазоне отдельно для каждой возрастной группы (если таких спортсменов нет, вывести соответствующее сообщение).

Класс Sportsman должен содержать закрытые поля: фамилия, год рождения, результаты соревнований (массив), сабскрипт для доступа к элементам массива результатов и все необходимые для решения задачи свойства и методы.

**Вариант 6.**

− Ввод с клавиатуры данных о спортсменах в массив объектов класса Sportsman.

− Вывод списка всех спортсменов с указанием года рождения в порядке возрастания среднего результата соревнований каждого спортсмена.

− Определение количества спортсменов для каждой возрастной группы (с одинаковым годом рождения), средний результат которых попал в заданный диапазон.

− Вывод списка спортсменов моложе заданного возраста (если таких спортсменов нет, вывести соответствующее сообщение).

Класс Sportsman должен содержать закрытые поля: фамилия, год рождения, результаты соревнований (массив), сабскрипт для доступа к элементам массива результатов и все необходимые для решения задачи свойства и методы.

**Вариант 7.**

− Ввод с клавиатуры данных о сотрудниках предприятия в массив объектов класса Employee.

− Вывод списка всех сотрудников предприятия с указанием отдела в алфавитном порядке.

− Определение количества сотрудников в каждом отделе, средняя зарплата которых больше средней зарплаты по всему предприятию.

− Вывод списка сотрудников, минимальная зарплата которых меньше заданной (если таких сотрудников нет, вывести соответствующее сообщение).

Класс Employee должен содержать закрытые поля: фамилия, наименование отдела, зарплата за последние n месяцев (массив), сабскрипт для доступа к элементам массива зарплат и все необходимые для решения задачи свойства и методы.

**Вариант 8.**

− Ввод с клавиатуры данных о сотрудниках предприятия в массив объектов класса Employee.

− Вывод списка всех сотрудников предприятия с указанием зарплаты за последние n месяцев в алфавитном порядке.

− Определение средней зарплаты по каждому отделу.

− Вывод списка сотрудников, которые получили зарплату меньше заданной больше одного раза (если таких сотрудников нет, вывести соответствующее сообщение).

Класс Employee должен содержать закрытые поля: фамилия, наименование отдела, зарплата за последние n месяцев (массив), сабскрипт для доступа к элементам массива зарплат и все необходимые для решения задачи свойства и методы.

**Вариант 9.**

− Ввод с клавиатуры данных о сотрудниках предприятия в массив объектов класса Employee.

− Вывод списка всех сотрудников предприятия в порядке возрастания средней зарплаты.

− Определение максимальной зарплаты по каждому отделу.

− Вывод списка сотрудников, которые средняя зарплата которых превышает среднюю по предприятию больше чем на 5% (если таких сотрудников нет, вывести соответствующее сообщение).

Класс Employee должен содержать закрытые поля: фамилия, наименование отдела, зарплата за последние n месяцев (массив), сабскрипт для доступа к элементам массива зарплат и все необходимые для решения задачи свойства и методы.

**Вариант 10.**

− Ввод с клавиатуры данных о преподавателях в массив объектов класса Lecturer.

− Вывод списка всех преподавателей университета в алфавитном порядке.

− Определение минимальной общей нагрузки за весь учебный год по каждой кафедре.

− Вывод списка преподавателей, у которых средняя нагрузка превышает среднемесячную по университету (если таких преподавателей нет, вывести соответствующее сообщение).

Класс Lecturer должен содержать закрытые поля: фамилия, кафедра, нагрузка в часах 10 месяцев (массив), сабскрипт для доступа к элементам массива нагрузки и все необходимые для решения задачи свойства и методы.

**Вариант 11.**

− Ввод с клавиатуры данных о преподавателях в массив объектов класса Lecturer.

− Вывод списка всех преподавателей университета с указанием годовой нагрузки каждого преподавателя, отсортированного по названию кафедры.

− Определение средней нагрузки за первые два месяца учебного года по каждой кафедре.

− Вывод списка преподавателей, у которых нагрузка хотя бы в одном месяце превышает среднемесячную нагрузку по университету (если таких преподавателей нет, вывести соответствующее сообщение).

Класс Lecturer должен содержать закрытые поля: фамилия, кафедра, нагрузка в часах 10 месяцев (массив), сабскрипт для доступа к элементам массива нагрузки и все необходимые для решения задачи свойства и методы.

**Вариант 12.**

− Ввод с клавиатуры данных о преподавателях в массив объектов класса Lecturer.

− Вывод списка всех преподавателей университета в алфавитном порядке с указанием максимальной месячной нагрузки

− Определение количества преподавателей по каждой кафедре с годовой нагрузкой ниже средней нагрузки за год по университету.

− Вывод списка преподавателей, у которых годовая нагрузка превышает среднюю годовую нагрузку по университету меньше чем на 10%(если таких преподавателей нет, вывести соответствующее сообщение).

Класс Lecturer должен содержать закрытые поля: фамилия, кафедра, нагрузка в часах 10 месяцев (массив), сабскрипт для доступа к элементам массива нагрузки и все необходимые для решения задачи свойства и методы.

**Вариант 13.**

− Ввод с клавиатуры данных о поездах в массив объектов класса Train.

− Вывод информации обо всех поездах в порядке возрастания номеров с указанием наличия свободных мест в заданном

− Определение количества свободных мест в поездах, следующих до заданной станции.

− Вывод списка поездов, отправляющихся из заданного города, в которых есть места для группы из заданного количества пассажиров (если таких поездов нет, вывести соответствующее сообщение).

Класс Train должен содержать закрытые поля: номер поезда, пункт отправления, пункт назначения, количество свободных мест в каждом вагоне (массив), сабскрипт для доступа к элементам массива свободных мест и все необходимые для решения задачи свойства и методы.

**Вариант 14.**

− Ввод с клавиатуры данных о поездах в массив объектов класса Train.

− Вывод информации о поездах в порядке убывания номеров с указанием наличия свободных мест в каждом

− Определение общего количества свободных мест по каждому пункту назначения.

− Вывод списка поездов, следующих в заданный город, в которых нет свободных мест (если таких поездов нет, вывести соответствующее сообщение).

Класс Train должен содержать закрытые поля: номер поезда, пункт отправления, пункт назначения, количество свободных мест в каждом вагоне (массив), сабскрипт для доступа к элементам массива свободных мест и все необходимые для решения задачи свойства и методы.

**Вариант 15.**

− Ввод с клавиатуры данных о поездах в массив объектов класса Train.

− Вывод информации о поездах номеров с указанием наличия свободных мест в каждом вагоне в порядке убывания общего количества свободных мест.

− Определение количества свободных мест в каждом поезде по заданному пункту отправления.

− Вывод списка поездов, следующих в заданный город, в которых есть свободные места, с указанием количества вагонов со свободными местами (если таких поездов нет, вывести соответствующее сообщение).

Класс Train должен содержать закрытые поля: номер поезда, пункт отправления, пункт назначения, количество свободных мест в каждом вагоне (массив), сабскрипт для доступа к элементам массива свободных мест и все необходимые для решения задачи свойства и методы.