## Итоговое задание по курсу «Объектно-ориентированное программирование с использованием языка C++»

С использованием STL разработать три приложения для решения следующих задач в консольных приложениях с использованием меню.

### Задача 1 (обязательная, максимум 8 баллов)

В векторе, состоящем из п целых элементов, выполнить следующие операции (по командам меню):

1. количество элементов вектора, равных 0
2. сумму элементов вектора, расположенных после минимального элемента
3. удалить элементы, встречающиеся менее двух раз
4. упорядочить вектор по возрастанию модулей элементов
5. продублировать отрицательные элементы вектора
6. сохранить вектор в бинарном файле при помощи потока вывода
7. прочитать вектор из бинарного файла при помощи потока ввода

### Задача 2 (для претендентов на 10 баллов).

С использованием классов и алгоритмов STL, лямбда-выражений написать приложение для решения следующей задачи из практикума Павловской Т.А. и Щупака Ю.А. по ООП на C++:

Написать приложения для учета книг в библиотеке. Сведения о книгах содержат: идентификатор (учетный номер единицы хранения), фамилию и инициалы автора, название книги, год издания, количество экземпляров данной книги в библиотеке. Сведения о библиотеке – адрес библиотеки, фамилия и инициалы директора, коллекция книг. Данные о библиотеке и книгах хранить в текстовом файле в формате CSV, символ разделителя «;».

Формат файла – в первой строке данные библиотеки, в остальных строках данные о книгах, одна книга – одна строка, начальная инициализация при первом запуске приложения – не менее 12 книг. С помощью меню реализовать выбор и выполнение следующих функций:

* Вывод данных библиотеки и коллекции книг в консоль
* Добавление книги, ввод данных с клавиатуры, сохранение измененной коллекции в текстовый файл
* Удаление книг по идентификатору, сохранение измененной коллекции в текстовый файл
* Изменение количества книг заданного автора на вводимое с клавиатуры значение, сохранение измененной коллекции в текстовый файл
* Отбор книг в отдельный список по заданному году издания, вывод этого списка в консоль, упорядочив по году издания, не изменять исходную коллекцию книг, файл данных
* Отбор заявок в отдельный список по автору, вывод этого списка в консоль, упорядочив по автору, не изменять исходную коллекцию книг, файл данных
* Сортировка книг по убыванию количества книг, вывод в консоль без изменения файла данных
* Сортировка книг по идентификатору, вывод в консоль без изменения файла данных
* Сортировка книг по названию, вывод в консоль без изменения файла данных

Постарайтесь свести к минимуму использование циклов – максимально используйте **алгоритмы STL**.

### Задача 3 (для претендентов на 12 баллов) – вариант 1

Класс **Bill** представляет собой разовый платеж за телефонный разговор. Класс должен включать поля: дата и время начала разговора, дата и время окончания разговора.

Время разговора вычисляется в минутах, неполная минута, считается за полную. Должен быть реализован метод, вычисляющий сумму к оплате за разговор.

Для представления даты и времени используйте класс **DateTime** с полями – день, месяц, год, час, минута, секунда. Реализуйте методы вычисления разности между двумя моментами времени в секундах. Перегрузите операции сравнения двух моментов времени.

Разработайте класс – список плательщиков ListPayer. Класс содержит список плательщиков за телефонные услуги, дату создания списка, номер списка.

Один элемент списка включает: информацию о плательщике – номер телефона, фамилия и инициалы, фамилия и инициалы плательщика, номер телефона, тариф за минуту разговора, скидка (в процентах), коллекция записей класса **Bill**, дату платежа, сумму платежа.

Реализовать для класса **ListPayer**:

* Метод добавления плательщика в список
* Метод удаления плательщика из списка
* Метод поиска плательщика по номеру телефона, найденные записи помещать в отдельную коллекцию, сохранять в бинарном потоке вывода
* Метод поиска плательщика по фамилии и инициалам, найденные записи помещать в отдельную коллекцию, сохранять в бинарном потоке вывода
* Метод поиска плательщиков по дате платежа, найденные записи помещать в отдельную коллекцию, сохранять в бинарном потоке вывода
* Метод вычисления полной стоимости платежей всего списка

Реализовать создание объектов **ListPayer**, операции объединения и пересечения списков.

* Операция объединения списка – бинарная операция, на выходе получаем список из элементов двух списков, без дубликатов.
* Операция пересечения списка – бинарная операция, на выходе получаем список из элементов имеющихся и в первом, и во втором списках, без дубликатов.

Реализовать отбор данных из списка плательщиков:

* По дате платежа, сортировка по номеру телефона
* По номеру телефона, сортировка по номеру телефона

Реализовать сортировку списка плательщиков:

* По номеру телефона
* По убыванию суммы к оплате
* По убыванию времени разговоров

### Задача 3 (для претендентов на 12 баллов) – вариант 2

Вы участвуете в разработке приложения для фитнес-клуба, занимающегося проведением тренировок в собственных залах под руководством тренеров или самостоятельно клиентами. Предприятие также может предоставлять свои залы в аренду для работы сторонних тренеров. Про работу предприятия с точки зрения разработки приложения известно следующее.

Можно выделить три категории пользователей: клиенты, тренеры, администрация (технический персонал оставляем без внимания 😊). Также в фитнес-клубе есть три зала (тренажерный, зал аэробики, зал боевых искусств).

Клиенты могут получать информацию о расписании групповых занятий, о занятости залов, тренеров индивидуальных занятий. Клиентом является любая персона, купившая абонемент. Клиент может зарегистрироваться на тренировку в группе или у тренера персонально. Если на тренировку зарегистрировалось менее 2х клиентов, тренировка отменяется, об этом извещается тренер и зарегистрировавшийся клиент.

Тренеры сообщают свой желаемый график администрации: желаемое время занятий и наиболее подходящий зал.

Администрация занимается приемом на работу и увольнением тренеров, продажей абонементов клиентам, составлением расписаний тренеров, организацией хозяйственной деятельности. Также администратор может формировать определенные отчеты при помощи приложения (например, отчет по тренерам – количество тренировок каждого тренера за определенный месяц).

Клиенты и персонал фитнес-клуба (пользователи приложения) характеризуются именем, фамилией и отчеством, номером паспорта. Персонал также характеризуется должностью (тренер, администратор, уборщик, …), окладом. Администратор только один, он контролирует нескольких тренеров.

После составления расписания или при назначении индивидуальной тренировки администратор уведомляет тренеров, клиентов индивидуальных тренировок. Также администратор извещает об изменении расписания всех клиентов фитнес-клуба.

Тренировка описывается местом и временем проведения, тренером, списком участников (для индивидуальной тренировки в списке только один участник).

Для 7 клиентов, 3 тренеров и 1 администратора разработать и протестировать следующую часть приложения: назначение тренеров на две групповые и одну индивидуальную тренировки – на одно и то же время, в один и тот же день, в разные залы. При этом на одно из групповых занятий записывается только один клиент (для демонстрации отмены тренировки). Используем STL и, конечно же лямбда-выражения. Начальное заполнение данных – из фиксированного набора.

## Варианты для выполнения задания

**Вариант 1:** Болдырев А., Горячев М., Сапелкин Д.

**Вариант 2:** Вагнер В., Куцый Д., Поддубный А.

## Дополнительно

Срок выполнения задания – 3 недели, до 16.08.2021.

Запись итогового занятия можно скачать [**по этой ссылке**](https://cloud.mail.ru/public/a564/FURgyENWz).