

Тестовое задание для Java программиста

Тестовое задание необходимо для проверки логики, умения проектировать и кодировать алгоритмы, навыков декомпозиции и ООП, презентации стиля написания кода (при удаленном тестировании отправить задачу по email или WhatsApp и получить реализацию). Время выполнения замерить.

Реализовать класс `APPDataParser` с одним `public` методом `normalizeGrafic(String [][] grafic): String[][]`, который на вход принимает двумерный массив строк с расписанием врачей и возвращает обработанный двумерный массив строк. Вложенные массивы — это расписание для одного врача на неделю. Первый элемент каждого вложенного массива это ФИО врача, далее идут семь строк с диапазонами рабочего времени по дням недели в формате `чч:мм-чч:мм`. Время может быть пустой строкой. Допускается, что по одному врачу может быть несколько расписаний с разными отрезками времени. Например,

```
arr = [  
    ["ИВАНОВ ИВАН ИВАНОВИЧ", '12:00-16:00', '12:00-16:00', '12:00-16:00', '12:00-16:00', '12:00-16:00', '12:00-16:00', ""],  
    ["ИВАНОВ ИВАН ИВАНОВИЧ", '08:00-12:00', '08:00-10:00', '08:00-16:00', "", "", ""],  
    ["ПЕТРОВ ПЕТР ПЕТРОВИЧ", '12:00-16:00', "", '12:00-16:00', "", '12:00-16:00', "", ""],  
    ...  
]
```

Метод должен обработать массив так, чтобы исключить повторения врачей, то есть если на одного врача несколько записей, то нужно объединить их в одну запись, объединяя при этом рабочее время по дням недели. Если объединяемые отрезки времени стыкуются или накладываются, то они должны образовать общий отрезок времени, если не пересекаются, то должны соединиться через запятую. Например, `'10:00-12:00' + '12:00-18:00' = '10:00-18:00'`, еще пример, `'10:00-14:00' + '12:00-18:00' = '10:00-18:00'`, еще пример, `'10:00-11:00' + '12:00-18:00' = '10:00-11:00,12:00-18:00'`, еще пример, `'10:00-14:00' + "" = '10:00-14:00'`

Реализация может содержать любое количество классов и методов. Приветствуется дополнительное описание в диаграммах UML и javadoc