**Практическое задание 10**

**Выполнение работ по отладке программы с использованием специализированных средств отладки**

**Цель работы:** совершенствовать умения разработки программы согласно требованиям проектной документации, совершенствовать умения тестирования программного кода.

**Оборудование:** ПК, IntelliJ IDEA

Источники:

1. Руководство Java. <https://metanit.com/java/tutorial/>
2. [Массивы Java: объявление, создание, заполнение и инициализация, определение размера, вывод на экран, методы и примеры (sky.pro)](https://sky.pro/media/massivy-java/?ysclid=lpplhkykdq723199542)
3. [Методы Arrays fill, copyOf, copyOfRange, sort, binarySearch в Java (javarush.com)](https://javarush.com/quests/lectures/questsyntaxpro.level05.lecture07)
4. [Класс Arrays в Java (javarush.com)](https://javarush.com/groups/posts/1933-klass-arrays-i-ego-ispoljhzovanie)
5. [Перевернуть массив в Java (techiedelight.com)](https://www.techiedelight.com/ru/reverse-array-java/)
6. Введение в массивы. <https://youtu.be/_mGgR5OhI6I>
7. Пример копирования массивов Java. <https://youtu.be/DaVBjoTrgH8>

**ЗАДАНИЯ К ПРАКТИЧЕСКОМУ ЗАНЯТИЮ**

**Для решения задачи используйте методические указания уроков 5, 6, 7**

Вам представлено описание библиотеки, которая позволит вернуть список свободных временных интервалов (заданного размера) в графике сотрудника.

Изучите алгоритм работы библиотеки. На основании описания параметров библиотеки подготовьте инструменты тестирования библиотеки.

Вход:

* список занятых промежутков времени (в двух массивах: startTimes - начало, durations - длительность),
* минимальное необходимое время для диагностики/консультации/работы (consultationTime),
* рабочий день сотрудника (начало - beginWorkingTime и завершение - endWorkingTime)

Выход:

* список подходящих свободных временных промежутков (в массив строк формата HH:mm-HH:mm)

**Пример:**

|  |  |
| --- | --- |
| Вход | Выход |
| startTime | duration  10:00 60  11:00 30  15:00 10  15:30 10  16:50 40  Working Times  08:00-18:00  Consultation Time  30 | 08:00-08:30  08:30-09:00  09:00-09:30  09:30-10:00  11:30-12:00  12:00-12:30  12:30-13:00  13:00-13:30  13:30-14:00  14:00-14:30  14:30-15:00  15:40-16:10  16:10-16:40  17:30-18:00 |

**Требования к программному коду:**

1. **Программа должна быть построена по принципу ООП. Иметь метод Main в котором должна быть указана информация о программе в форме:**

Версия программы

Функции программы

Данные о разработчиках

©️ все права защищены

1. **Программа должна иметь класс** Calculations

|  |  |
| --- | --- |
| Название класса | Calculations |
| Название метода | availablePeriods() |
| Входящие обязательные параметры | LocalTime[] startTimes,  int[] durations,  LocalTime beginWorkingTime,  LocalTime endWorkingTime,  int consultationTime |
| Возвращаемые параметры | string[] |

1. **Программа должна иметь спецификацию сценариев тестов для метода** availablePeriods**; сценарии должны** **предусматривать различные ситуации.** Например, недостаточное время в промежутках между ранее созданными консультациями, либо в начале рабочего дня, либо в конце рабочего дня; различная длительность консультация и т.д.

Шаблон спецификации тестов сценария

|  |  |
| --- | --- |
| **Наименование** | **Описание** |
| Наименование  проекта | ConsoleApp32 |
| Номер версии | 1.0 |
| Имя тестера | Шишкина Н.В. |
| Даты тестирования | 12.05.2023 8:45 |
| Test Case # | Тест #01 (RectangleArea\_3and5\_15returned) – правильность вычисления функции a\*b |
| Приоритет теста (низкий/средний/высокий) | высокий |
| Наименование варианта тестирования | RectangleArea\_3and5\_15returned |
| Резюме испытания | Правильно вычисленное значение |
| Шаги тестирования | * Задать a=3; b=5 * Вычислить правильное значение expected = a\*b * Вызвать тестируемую функцию и вычислить int actual = g.RectangleArea(a, b); * Сравнить результаты вычисления с результатами программы (expected, actual) * Оценить результат |
| Тестовые данные | Вводятся внутри кода теста  a=3; b=5 |
| Ожидаемый  результат | 15 |
| Фактический  результат | 15 |
| Предусловия | Задать правильные параметры расчета |
| Постусловия | - |
| Статус (Pass/Fail) | Pass |
| Комментарии |  |

1. **Реализуйте спецификацию тестов с использованием Junit тестов. Тесты должны быть интегрированы ы код в отдельном пакете.**
2. **Докажите правильность работы программного кода.**
3. **Оформите отчет:**

|  |  |
| --- | --- |
| Класс | Листинг |
| Main |  |
| Calculations |  |
| Test |  |