

Практика 1: Введение, потоки, продвинутый JUnit

16.02.2018г

1. Формальности

Итак, начинается вторая часть изучения программирования на примере языка Java. Пары в этом семестре будут логическим продолжением того, что было в предыдущем семестре — немножко больше многопоточности, программирования сетевых приложений, многопоточных И сетевых приложений, приложений с пользовательским интерфейсом наконец-то, и даже немного веб-приложений, если мы до них дойдём. Как я понял, так или иначе почти всё затрагивалось при программировании под android, но в этот раз всё будет более системным и обстоятельным.

Сначала, как водится в начале семестра, формальности. В конце будет экзамен, оценка за который получается, в том числе, и из оценки по практике, так что в этом семестре будет не просто зачёт/незачёт, а некоторое число. Чтобы получить оную оценку, надо, как обычно, качественно и вовремя делать домашки, писать контрольные, сдавать их через HwProj¹, делая пуллреквест в свой репозиторий и кидая в HwProj ссылку на пуллреквест. Домашек будет меньше, чем в прошлом семестре, но они будут более объёмными. Будет три больших работы (своя система контроля версий, свой FTP-сервер и свой torrent-клиент) разбитых на несколько заданий, поэтому если не сделать вовремя начало, то не получится сдать и конец, так что лучше не запускать. При сдаче домашки в любой момент можно “сдаться” и не исправлять замечания, тогда вы получите столько баллов, на сколько насадали, но можно исправить и получить больше. Списывать, как обычно, нельзя, и вообще, имеет смысл стараться решать задачи самостоятельно, курс ориентирован на сольное прохождение, к тому же, если что — всегда можно спросить у меня.

¹ Система сдачи домашних заданий HwProj, URL: <http://hwproj.me/> (дата обращения: 13.02.2017г.)

Напомню про то, за что снимались баллы в прошлом семестре:

Пропущенный дедлайн	баллы делятся на два
Задача на момент дедлайна не реализует все требования условия	пропорционально объёму невыполненных требований
Неумение пользоваться гитом	-2
Проблемы со сборкой (в том числе, забытый <code>org.jetbrains.annotations</code>)	-2
Отсутствие JavaDoc-ов для всех классов, интерфейсов и публик-методов	-2
Отсутствие описания метода в целом	-1
Слишком широкие области видимости для полей	-2
<code>if (...) return true; else return false;</code>	-2
Именованые классов-полей-методов-... и прочие code conversions	-1
Неиспользование <code>try-with-resources</code> там, где это было бы уместно	-1
Комментарии для параметров с заглавной буквы	-0.5

Обнаружение ошибок из этого списка сразу влечёт снятие баллов за задачу, даже без права исправить ошибку. На остальные ошибки я буду указывать и будет возможность их поправить, но наиболее распространённые ошибки будут пополнять этот список, так что имеет смысл ходить на пары, чтобы вовремя узнать, что меня в очередной раз ужаснуло и заставило расширить список “плохих” ошибок. Табличку с оценками я выложу на вики.