



Втора лабораториска вежба по Софтверско инженерство

Рок: 31.05.2022 (вторник), 23:59

Упатство за изработка

- Креирајте Ваш оддалечен (remote) репозиториум на Github што ќе биде именуван во форматот **SI_2022_lab2_INDEX**. (INDEX да биде вашиот број на индекс). Пример: **SI_2022_lab2_111111**. Креирајте го репозиториумот со истиот профил кој го користевте за првата лабораториска вежба. Репозиториумот направите го **приватен**, а по 31.05.2022, ќе го направите јавен. При креирање на репозиториумот, изберете ја опцијата *Add a README file*. Во README документот наведете Ваше име, презиме и број на индекс.
- Во README документот внесувајте ги одговорите на барањата 2-5.
- **Барањето 2 (Control Flow Graph) мора да биде изработено за да се признае вежбата за прикачена и изработена.**

Барања:

1. Преземете го кодот во .java формат од [тука](#). Променете го името на фајлот во SIlab2.java, доколку не е зачувано така.

За кодот во функцијата **function** потребно е да ги изработите задачите од следните прашања. Оваа функција прима еднодимензионална листа со должина n^2 која ја интерпретира како дводимензионална $n \times n$ табла. Листата се состои од нули и # што означуваат бомби. Целта на функцијата е да врати листа во која што на секоја позиција каде што има нула ќе стои бројот на бомби лево, десно, над и под соодветната позиција.

Пример: листата ["0", "#", "0", "0", "0", "#", "0", "#", "#"] се интерпретира како

```
0 # 0
0 0 #
0 # #
```

и за овој влез функцијата треба да врати ["1", "#", "2", "0", "3", "#", "1", "#", "#"]

2. За дадениот код во функцијата **function** нацртајте Control Flow Graph со некоја алатка за цртање дијаграми. Внесете го нацртанот CFG во документацијата.
3. Пресметајте ја цикломатската комплексност на дадениот код. Објаснете како стигнавте до резултатот.
4. Напишете ги сите тест случаи според Every Statement критериумот. Напишете и објаснете ги тест случаите во документацијата.
5. Напишете ги сите тест случаи според Every Branch критериумот. Напишете и објаснете ги тест случаите во документацијата.



6. Креирајте Gradle build за обична Java апликација. Проектот треба биде именуван SI_lab2_{број на индекс} (пр. SI_lab2_111111).

Во src/main/java вметнете ја класата што ја превземавте во првото прашање. Мора да биде потпишана SIlab2.

Во src/test/java креирајте класа SIlab2Test во која што ќе напишете две функции (по една за секој критериум за тестирање) за unit тестирање (ќе имаат нотација @Test). Во функциите за тестирање, со помош на assert изрази, проверете ги сите тест случаи што ги утврдите во претходните прашања.

Проектот прикачете го на GitHub репозиториумот.

7. По завршување на сите задачи, репозиториумот треба да го содржи Java проектот во кој се напишани тестовите и документација во која што ќе има:

- слика од нацртаниот CFG
- пресметка за цикломатската комплексност на функцијата и објаснување за дадениот резултат
- утврдените тест случаи според доделените критериуми за тестирање
- објаснување како ги напишавте unit тестовите за секој од критериумите

Пример за тоа како треба да изгледа целиот проект имате [тука](#).