

Лабораторна робота 2

Модульне тестування. Ознайомлення з засобами та практиками модульного тестування

Мета лабораторної роботи – навчитися створювати модульні тести для вихідного коду розроблювального програмного забезпечення.

Завдання

1. Додати до проекту власної узагальненої колекції (застосувати виконану лабораторну роботу №1) проект модульних тестів, використовуючи певний фреймворк (Nunit, Xunit, тощо).
2. Розробити модульні тести для функціоналу колекції.
3. Дослідити ступінь покриття модульними тестами вихідного коду колекції, використовуючи, наприклад, засіб AxoCover.

Варіанти

Номер варіанта	Опис узагальненої колекції
1	Стек
2	Черга
3	Бінарне дерево
4	Дек (черга з двома кінцями)
5	Динамічний масив
6	Словник
7	Відсортований динамічний масив
8	Кільцевий список
9	Динамічний масив з довільним діапазоном індексу
10	Зв'язаний список

Контрольні запитання

1. Що таке модульне тестування?
2. Як використовуються модульні тести?
3. Назвіть вимоги до юніт-тестів.
4. Наведіть переваги використання юніт-тестів у розробці ПЗ.
5. Що таке рефакторинг?
6. Які метрики та засоби використовуються для оцінювання ефективності застосування юніт-тестів у проекті?
7. Що таке TDD? Назвіть переваги застосування TDD.
8. Що таке принцип «Triple A»? Поясніть сутність його використання.
9. Як використовуються в юніт-тестах класи Assert?
10. Що таке Mock та Stub? З якою ціллю вони використовуються в юніт-тестах?

Список літератури

1. Рой Ошероув Искусство автономного тестирования с примерами на С# (Roy Osherove The Art of Unit Testing With Examples in .NET).
2. <https://github.com/nunit/docs/wiki/NUnit-Documentation>
3. <https://msdn.microsoft.com/ru-ru/library/ms182532.aspx>
4. <https://habrahabr.ru/post/176137/>