

СТРУКТУРА, ЗМІСТ ТА ОФОРМЛЕННЯ ЗВІТУ

За результатами виконання лабораторної роботи студентом самостійно готується звіт, який повинен мати наступну структуру і зміст:

- титульний аркуш;
- мета роботи, номер варіанту й завдання;
- самостійно одержані результати виконання завдань, перелік яких визначено порядком виконання лабораторної роботи;
- висновки;
- додатки: тест-сьюти TS_lab1_task_1, TS_lab1_task_2.

Оформлення звіту повинне відповідати наступним вимогам:

- спосіб виконання - машинописний (комп'ютерний);
- параметри сторінки:
 - формат А4;
 - верхнє, нижнє, праве поля – 15 мм, ліве поле – 20 мм;
- текст:
 - шрифт Times New Roman;
 - розмір шрифту – 12 пт;
 - міжабзацний інтервал – 0 пт;
 - міжрядковий інтервал – 1 пт;
 - абзацний відступ – 15 мм;
- лістинг (текст) програм:
 - шрифт Courier New;
 - розмір шрифту – 10 пт;
 - міжабзацний інтервал – 0 пт;
 - міжрядковий інтервал – 1 пт.

Виконане завдання лабораторної роботи та підготовлений звіт безпосередньо під час заняття дає студентові право подати його до захисту в електронній формі (файл).

САМОСТІЙНА ПІДГОТОВКА ДО ВИКОНАННЯ ЛАБОРАТОРНОЇ РОБОТИ № 1

ПЕРЕЛІК ТЕОРЕТИЧНИХ ПОЛОЖЕНЬ ДИСЦИПЛІНИ

Для допуску до виконання лабораторної роботи № 1, мета якої полягає у підтвердженні окремих теоретичних положень навчальної дисципліни “Модульне програмування” (лекція 1, СРС 1) під час лабораторного заняття, студент **повинен знати** наступне:

- процеси життєвого циклу програмних засобів відповідно до міжнародного стандарту ISO/IEC 12207:2008 “Systems and software engineering - Software life cycle processes”;
- поняття абстрагування, інкапсуляції, модуля;
- сутність, методологія, порядок (алгоритм) реалізації модульного проектування програмних засобів в межах модульного програмування як парадигми;
- метод функціональної декомпозиції;
- стадії утворення модульного програмного забезпечення;
- способи й нотації документування архітектури й алгоритмів програмного забезпечення, яке реалізовується;
- мову програмування Pascal (Free Pascal);
- порядок проектування графічних додатків та використання візуальних компонентів у кросплатформовому Lazarus IDE;
- стандартні модулі FPC (на базі Lazarus IDE) для оброблення символьних і рядкових даних, математичних обчислень та порядок їх використання під час реалізації модульного ПЗ;
- сутність і призначення тестування ПЗ, його види й рівні (класифікація за рівнем деталізації програмного засобу);
- тестові артефакти та способи їх документування;
- підходи до системного тестування ПЗ;
- сутність і склад тестового набору (Test Suite);
- призначення й структура тестового випадку (Test Case);
- види тест-кейсів за очікуваним результатом.

ПИТАННЯ ДЛЯ САМОКОНТРОЛЮ ПІДГОТОВЛЕНOSTI

ДО ВИКОНАННЯ ЛАБОРАТОРНОЇ РОБОТИ № 1

1. Які процеси реалізації програмних засобів визначені міжнародним стандартом ISO/IEC 12207 (розділ 7) та в чому полягає їх мета й сутність?
2. Що розуміють під артефактами процесів реалізації програмного забезпечення та їх документування?
3. Який порядок створення проекту графічного додатка (a graphical Free Pascal application using the cross-platform LCL library for its GUI) в Lazarus IDE?
4. Яким чином у Lazarus IDE додаються візуальні компоненти на форму додатка?
5. Яким чином можливо реалізувати введення і виведення даних у графічному додатку, що розробляється у Lazarus IDE?
6. Яке призначення візуальних компонентів (об'єктів) Label, Edit, Memo, Button?
7. Яким чином здійснюється доступ до властивостей об'єктів Label, Edit, Memo, Button (зокрема, даних, які вводять або виводяться у вікні графічного додатка)?
8. Яким чином здійснюється доступ до даних, введених користувачем у поля об'єктів Edit, Memo і якого вони типу?
9. Яке призначення та інтерфейс стандартних математичних, рядкових, календарних функцій, зокрема, sin, cos, tan, pow, sqrt, exp, log, abs, modf, random, length, copy, concat, delete, insert, pos, uppercase, SetLength, UTF8Length, UTF8Delete, UTF8Pos, LowerCase, UTF8LowerCase, UTF8UpperCase, CompareStr, CompareText, IntToStr, StrToInt, StrToFloat, FloatToStr, AnsiToUTF8, UTF8Decode, UTF8Encode, UTF8ToAnsi, DateToStr, DateTimeToStr, date, now, time?
10. Як у Lazarus IDE реалізовується обробник (процедура) натиснення кнопки (Button)?

11. Яким чином здійснюється виведення числових даних у вікні графічного додатка за допомогою компонентів Edit, Label?
12. Які стандартні функції забезпечують перетворення типів?
13. Яким чином можливо програмно отримати значення календаря операційної системи (дата, час) і у якому воно форматі?
14. Що здійснюється в процесі кваліфікованого тестування програмних засобів?
15. Яким чином реалізовується системне тестування (System Testing) програмного забезпечення?
16. В чому полягає сутність підходів до системного тестування підходами на основі вимог (Requirements Based) та на основі випадків використання (Use Case Based) ПЗ?
17. Як формується (описується) і який склад тест-кейса?
18. З яких атрибутів (полів) складається тестовий випадок та яке їх призначення?
19. Які можливі результати виконання тест-кейса (результат тестування / Test Result)?
20. Яким чином документується процес тестування ПЗ?

СПИСОК РЕКОМЕНДОВАНОЇ ЛІТЕРАТУРИ

1. International Standard ISO/IEC 12207:2008 “Systems and software engineering – Software life cycle processes”.
2. Бахтизин В. В. Технология разработки программного обеспечения : учеб. пособие / В. В. Бахтизин, Л. А. Глухова. – Минск : БГУИР, 2010. – 267 с.
3. Давыдов В. Г. Программирование и основы алгоритмизации: Учеб. пособие/ В. Г. Давыдов. – М.: Высш. шк., 2003. – 447 с.
4. Алексеев Е. Р. Free Pascal и Lazarus: Учебник по программированию / Е. Р. Алексеев, О. В. Чеснокова, Т. В. Кучер – М.: ALT Linux; Издательский дом ДМК-пресс, 2010. – 440с.

5. Мансуров К. Т. Основы программирования в среде Lazarus [Электронный ресурс] // К. Т. Мансуров. – 2010. – [772 с.] : Режим доступа: <http://ir.nmu.org.ua/bitstream/handle/123456789/84080/9166d95015a59c63a91f0a7df253d50a.pdf?sequence=1>. – Назва з екрану. – Дата звернення: 14.01.2016.
6. Ефлов В. Б. Программирование на языке Pascal в среде Lazarus: учебное пособие для студентов и преподавателей вузов / сост. : В. Б. Ефлов, Ю. В. Никонова. – Петрозаводск : Изд-во ПетрГУ, 2013. – 53 с.
7. Савин Р. Тестирование Дот Ком, или Пособие по жестокому обращению с багами в интернет-стартапах / Роман Савин. – М.: Дело, 2007. – 312 с.
8. Основы инженерии качества программных систем: 2-е изд., перераб. и доп. / [Андон Ф. И., Коваль Г. И., Коротун Т. М. и др.] – К.: Академперіодика, 2007. – 672 с.
9. Волкова С. О. Тестування як засіб підвищення якості та надійності системи діагностики / С.О. Волкова, О.М. Трунов // Авиационно-космическая техника и технология. – 2007. – № 9 (45). – С. 207-211.
10. Малюга В. Г. Аналіз та дослідження особливостей процедури проведення успішного тестування при розробці комплексу програм автоматизованої системи конструювання розкладу занять / В. Г. Малюга, О. М. Місюра, В. В. Калачова, О. А. Трублін // Системи обробки інформації. – 2015. – Вип. 12 (137). – С. 194-198.

ЛАБОРАТОРНА РОБОТА № 1

ПРОЦЕСИ, МЕТОДИ Й ЗАСОБИ РЕАЛІЗАЦІЇ МОДУЛЬНОГО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

Мета роботи — набуття практичних навичок застосування методів модульного проектування, засобів вільного кросплатформового середовища розроблення програм Lazarus, методики системного тестування під час реалізації модульного програмного забезпечення в межах міжнародного стандарту ISO/IEC 12207.

ЧАС ВИКОНАННЯ ЛАБОРАТОРНОЇ РОБОТИ

4 академічні години.

ОБЛАДНАННЯ, МАТЕРІАЛИ, ПРОГРАМНІ ЗАСОБИ

Для виконання лабораторної роботи необхідні:

- персональний комп'ютер з графічною операційною системою Windows або Linux;
- вільне кросплатформове середовище розроблення програмного забезпечення Lazarus;
- текстовий редактор (OpenOffice Writer, Microsoft Word або ін.);
- файл Artifact_TEST_SUITE_lab.doc, який розміщено на локальному сервері кафедри програмування та захисту інформації у теці дисципліни “Модульне програмування”.

ЗАВДАННЯ ДО ЛАБОРАТОРНОЇ РОБОТИ

1. Реалізувати програмний засіб (графічний додаток) розв'язування задачі 1.1, використовуючи стандартні модулі оброблення *символьних і рядкових* даних.
2. Реалізувати програмний засіб (графічний додаток) розв'язування задачі 1.2, використовуючи стандартні модулі *математичних* обчислень.

ПОРЯДОК ВИКОНАННЯ ЛАБОРАТОРНОЇ РОБОТИ ТА МЕТОДИЧНІ ВКАЗІВКИ

1. У викладача отримати умову задач 1.1 та 1.2 за варіантом.
2. До звіту з лабораторної роботи (далі – звіт) записати тему й мету роботи, номер варіанту, завдання.
3. Здійснити аналіз і постановку задачі 1.1.
4. Відповідно до міжнародного стандарту ISO/IEC 12207:2008 (підрозділ 7.1 розділу 7) до задачі 1.1 здійснити:
 - 4.1) аналіз вимог до програмного забезпечення;
 - 4.2) проектування архітектури модульного ПЗ;
 - 4.3) детальне проектування програмного засобу.
5. Одержані артефакти задокументувати й включити до звіту.
6. В теці дисципліни на локальному сервері кафедри відкрити файл Artifact_TEST_SUITE_lab.doc, зберегти його як Artifact_TEST_SUITE_lab_Прізвище_t1.doc у \MP\Прізвище_Lab1\, заповнити заголовок документа: поля Test Suite Description, Name of Project / Software, Level of Testing, де $N = 1$, *Прізвище* – прізвище автора тестового набору.
7. Розробити тестовий набір (Test Suite) TS_lab1_task_1 для здійснення системного тестування (System Testing) програмного забезпечення розв’язування задачі 1.1, який повинен складатись не менш як п’яти негативних тестових випадків (Test Case), створених за підходом на основі вимог (Requirements Based) до програмного засобу.
8. Описані тест-кейси за структурою Test Case ID → Test Case description → Author Test Case → Preliminary Steps → Action (test steps) → Expected Result → Test Result (passed/failed) → Implementer належно задокументувати у файлі Artifact_TEST_SUITE_lab_Прізвище_t1.doc (атрибути Test Result та Implementer залишити порожніми; вони заповнюються тестувальником за результатами виконання тест-кейсів).

9. В Lazarus IDE створити проект графічного додатка (a graphical Free Pascal application using the cross-platform LCL library for its GUI), іменувати його lab1_task1_Прізвище та зберегти у \MP\Прізвище_Lab1\.
10. На основі результатів проектування програмного забезпечення розв’язування задачі 1.1:
 - 10.1) налаштувати інтерфейс додатка: встановити розмір і розташування форми; додати, відповідно розмістити та налаштувати необхідні компоненти введення/виведення даних (Edit, Label, Memo тощо), подій/керування (кнопка Button) та ін.;
 - 10.2) мовою програмування реалізувати артефакти детального проектування ПЗ – розробити програмний(ні) модуль(і);
 - 10.3) скопіювати проект графічного додатка lab1_task1_Прізвище (Ctrl+F9).
11. Створений exe-файл (додаток lab1_task1_Прізвище.exe) скопіювати у \MP\Прізвище_Lab1\software\.
12. Відповідно до ISO/IEC 12207:2008 здійснити системне тестування створеного програмного забезпечення за розробленим тест-сюжетом TS_lab1_task_1.
13. Результати тестування належно задокументувати: за результатами виконання тестів заповнити атрибути/поля/ тест-кейсів Test Result та Implementer).
14. У випадку виявлення помилок у ПЗ за результатами виконання тест-кейсів TS_lab1_task_1 (стан “failed” або “blocked” атрибута “Test Result”), здійснити їх про аналіз і усунення, процес тестування повторити.
15. Лістинг (текст) розробленого програмного забезпечення розв’язування задачі 1.2 включити до звіту.
16. Здійснити аналіз і постановку задачі 1.2.
17. Згідно зі стандартом ISO/IEC 12207:2008 (підрозділ 7.1 розділу 7) до задачі 1.2 здійснити:
 - 17.1) аналіз вимог до програмного забезпечення;

- 17.2) проектування архітектури модульного ПЗ;
- 17.3) детальне проектування програмного засобу.
18. Одержані артефакти задокументувати й включити до звіту.
 19. В теці дисципліни на локальному сервері кафедри відкрити файл `Artifact_TEST_SUITE_lab.doc`, зберегти його як `Artifact_TEST_SUITE_lab_Прізвище_t2.doc` у `\MPV\Прізвище_Lab1\`, заповнити наступні поля заголовка документа: `Test Suite Description`, `Name of Project / Software`, `Level of Testing`, де $N = 2$, *Прізвище* – прізвище автора тестового набору.
 20. Розробити тестовий набір `TS_lab1_task_2` для здійснення системного тестування ПЗ розв'язування задачі 1.2, який повинен складатись не менш як з п'яти активних тест-кейсів, створених за підходом на основі випадків використання (Use Case Based) програмного засобу, який реалізовується.
 21. Розроблені тест-кейси належно задокументувати у файлі `Artifact_TEST_SUITE_lab_Прізвище_t2.doc` (крім атрибутів `Test Result` та `Implementer`).
 22. В Lazarus IDE створити проект графічного додатка, іменувати його `lab1_task2_Прізвище` та зберегти у `\MPV\Прізвище_Lab1\`.
 23. На основі результатів проектування програмного забезпечення розв'язування задачі 1.2:
 - 23.1) налаштувати інтерфейс додатка: встановити розмір і розташування форми; додати, відповідно розмістити та налаштувати необхідні компоненти (об'єкти);
 - 23.2) мовою програмування реалізувати артефакти детального проектування ПЗ – розробити програмний(ні) модуль(і);
 - 23.3) скопіювати проект графічного додатка `lab1_task2_Прізвище` (`Ctrl+F9`).
 24. Створений `exe`-файл (додаток `lab1_task2_Прізвище.exe`) скопіювати у `\MPV\Прізвище_Lab1\software\`.
 25. Здійснити системне тестування створеного додатка задачі 1.2 за розробленим тест-сюютом `TS_lab1_task_2`.

26. Результати тестування належно задокументувати: за результатами виконання тестів заповнити `Test Result` та `Implementer`.
27. У випадку виявлення помилок у ПЗ за результатами виконання тест-кейсів `TS_lab1_task_2`, здійснити їх про аналіз і усунення, процес тестування повторити.
28. Лістинг (текст) розробленого графічного додатка задачі 1.2 включити до звіту.
29. Тестові набори `TS_lab1_task_1`, `TS_lab1_task_2` включити до звіту як додаток.
30. Одержані результати виконання лабораторної роботи зберегти на носій (флеш-накопичувач, хмарне сховище даних тощо) задля їх подальшого використання під час виконання наступних лабораторних робіт.
31. Здійснити аналіз ходу виконання й одержаних результатів виконання лабораторних завдань.
32. Підготувати обґрунтовані висновки (підсумки)¹ з виконаної лабораторної роботи обсягом не менше ½ сторінки машинописного тексту та включити їх до звіту; у висновках рекомендовано також окремо зазначити особисті враження, вмотивовані пропозиції, рекомендації, застереження, зауваження тощо.
33. Затвердити у викладача розроблене програмне забезпечення (`exe`-файли у `\MPV\Прізвище_Lab1\software\`).
34. Підготувати звіт з лабораторної роботи № 1 відповідно до встановлених вимог щодо структури, змісту й оформлення.
35. Подати одержані результати виконання лабораторної роботи до захисту.

¹ висновки, як результат розумової діяльності студента, повинні, зокрема, містити стисле викладення здобутих в процесі виконання лабораторної роботи результатів, опис проблем, які виникали під час реалізації завдань, та шляхи їх вирішення; структура підсумків має бути логічною і охоплювати весь процес виконання лабораторної роботи тощо.