



МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

ХАРКІВСЬКИЙ НАЦІОНАЛЬНИЙ
УНІВЕРСИТЕТ РАДІОЕЛЕКТРОНІКИ

МЕТОДИЧНІ ВКАЗІВКИ

до виконання курсової роботи з дисципліни
"ОБ'ЄКТНО-ОРІЄНТОВАНЕ ПРОГРАМУВАННЯ"

Харків 2016

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

ХАРКІВСЬКИЙ НАЦІОНАЛЬНИЙ
УНІВЕРСИТЕТ РАДІОЕЛЕКТРОНІКИ

МЕТОДИЧНІ ВКАЗІВКИ

до виконання курсової роботи з дисципліни
"ОБ'ЄКТНО-ОРІЄНТОВАНЕ ПРОГРАМУВАННЯ"

для студентів усіх форм навчання
напряму 6.050103 – Програмна інженерія,
спеціальність 121 – «Інженерія програмного забезпечення»
(освітня програма «Програмна інженерія»)

ЗАТВЕРДЖЕНО

кафедрою ПІ.

Протокол № 15 від 1.05.2016 р.

Харків 2016

Методичні вказівки до виконання курсової роботи з дисципліни «Об'єктно-орієнтоване програмування» для студентів усіх форм навчання напрямку 6.050103 – Програмна інженерія, спеціальність 121 – Інженерія програмного забезпечення (освітня програма «Програмна інженерія») / Упоряд.: В.М. Бондарєв, Ю.Ю. Черепанова. – Харків: ХНУРЕ, 2016. – 40 с.

Упорядник: В.М. Бондарєв, Ю.Ю. Черепанова

Рецензент: О.А. Галуза, професор каф. КМММ НТУ «ХП»

ЗМІСТ

ЗАГАЛЬНІ ПОЛОЖЕННЯ	4
1 МЕТА КУРСОВОГО ПРОЕКТУВАННЯ	4
2 ТЕМАТИКА КУРСОВИХ РОБІТ.	5
3 ПОРЯДОК ВИКОНАННЯ КУРСОВОЇ РОБОТИ.	6
3.1 Формулювання вимог до програми	6
3.2 Проектування.....	6
3.3 Кодування програми	7
3.4 Тестування програми	7
3.5 Пояснювальна записка	7
4 СКЛАД ПОЯСНЮВАЛЬНОЇ ЗАПИСКИ.....	8
5 ВИМОГИ ДО ОФОРМЛЕННЯ КУРСОВОЇ РОБОТИ.....	9
5.1 Загальні вимоги	9
5.2 Вимоги до тексту.....	9
5.3 Нумерація.....	11
5.4 Перелічення.....	11
5.5 Оформлення таблиць.....	12
5.6 Оформлення рисунків.....	13
5.7 Оформлення частин програмного коду.....	14
5.8 Складання переліку посилань.....	15
5.9 Додатки.....	16
6 ОРГАНІЗАЦІЯ ПРОЕКТУВАННЯ І ЗАХИСТ РОБОТИ.....	16
ПЕРЕЛІК ПОСИЛАНЬ.....	17
Додаток А – Приклади тем курсових робіт.....	18
Додаток Б – Узгодження щодо кодування програм.....	25
Додаток В – Аркуш завдання та календарний план	31
Додаток Г – Зразок титульного аркуша	33
Додаток Д – Зразок оформлення реферату	34
Додаток Е – Приклад оформлення переліку посилань.....	35
Додаток Ж – Зразок оформлення тексту пояснювальної записки	36
Додаток И – Приклад складання змісту	40

ЗАГАЛЬНІ ПОЛОЖЕННЯ

Метою дисципліни «Об'єктно-орієнтоване програмування» є вивчення основних принципів об'єктно-орієнтованого програмування, та набуття навичок розробки об'єктних програм. Дисципліна має зосередити студента на найбільш важливих рисах програмування, які не залежать від таких обставин, як тип процесора або операційна система. Об'єктно-орієнтоване програмування дуже слушно надає таку можливість, бо воно не залежить навіть від мови програмування.

За результатом вивчення дисципліни студенти повинні:

а) знати загальні принципи об'єктно-орієнтованого програмування, реалізацію об'єктної моделі мови програмування, архітектуру програм з графічним інтерфейсом користувача;

б) вміти створювати програми в об'єктно-орієнтованій парадигмі, доцільно використовувати в програмах можливості обраної мови програмування; використовувати бібліотечні класи для розробки графічних та консольних застосунків із загальними засадами об'єктно-орієнтованого проектування;

в) володіти методами та технологіями об'єктно-орієнтованого програмування.

Працюючи над курсовою роботою, студент повинен вивчити окремі фази розробки програмного забезпечення і навчитися поєднувати їх в одне ціле – в свій проект. І хоч неможливо очікувати, що за той обмежений час, який є у студента, він створить справжній програмний продукт, викладач повинен максимально наблизити його роботу до реальних умов.

1 МЕТА КУРСОВОГО ПРОЕКТУВАННЯ

Метою курсового проектування є закріплення знань, набутих у ході вивчення дисципліни «Об'єктно-орієнтоване програмування», а головне, набуття навичок у розробці програмних продуктів.

Крім того, під час захисту курсової роботи студент повинен проявити такі властивості як впевненість у власних знаннях, вміння відстоювати власну думку, вміння виступати перед аудиторією.

2 ТЕМАТИКА КУРСОВИХ РОБІТ

Для курсового проектування обираються теми різної проблематики. Приклади тем курсових робіт наведені у додатку А. Приклади тем взяті з книги [3] та доопрацьовані.

Студент повинен підтвердити вибір теми, здавши викладачеві заповнений аркуш завдання (див. додаток В). Якщо студент не пропонує своєї теми в наданий для цього термін, викладач надає йому власну тематику. Додатково теми фіксуються у журналі групи на відповідній сторінці. Після цього тему змінити не можна.

Незалежно від теми програма має відповідати таким загальним вимогам (вимоги до коду програми наведені в додатку Б):

а) *Стійкість програми.* Програма не має втрачати працездатності за жодних, навіть некоректних, дій користувача. Будь-які дії, які загрожують втратою інформації, мають бути підтверджені користувачем. Введена інформація всюди, де це можливо, має піддаватися логічному контролю;

б) *Забезпечення цілісності даних.* За будь-яких дій користувача не повинні губитися дані або їх цілісність (некоректність індексів, втрата посилань в зв'язках після видалення-додавання записів тощо);

в) *Функціональна повнота.* Мають бути реалізовані всі функції, зазначені в специфікації програми. Обов'язково реалізувати функції додавання, редагування даних;

г) *Терміни та інтерфейс.* У діалогових вікнах мають використовуватися тільки терміни, зрозумілі користувачеві, і не мають використовуватися терміни розробника («запис», «індексація» тощо). Поява службових англійських повідомлень неприпустима. У повідомленнях користувачеві слід дотримуватися норм ввічливості, кольорова гамма має слідувати загальноприйнятим рекомендаціям;

д) *Використання клавіатури.* На будь-якому етапі натискання будь-якої клавіші має ігноруватися або викликати передбачені дії, описані в засобах допомоги. Прив'язка дій до клавіш має бути загальноприйнятою: F1 – допомога; Enter – згода, завершення введення; Esc – відмова, повернення до попереднього вузла алгоритму (з відновленням екранної форми); Tab – перехід до наступного поля, вікна; Shift-Tab – повернення до попереднього поля та ін.

3 ПОРЯДОК ВИКОНАННЯ КУРСОВОЇ РОБОТИ

Основні етапи виконання курсової роботи такі:

- формулювання вимог до програми;
- загальне проектування;
- детальне проектування;
- кодування;
- тестування;
- оформлення пояснювальної записки;
- захист роботи.

Пункти 3 – 5 повторюють багаторазово по відношенню до кожної частини програми. Частини виділяють на етапі загального проектування.

3.1 Формулювання вимог до програми

На етапі формулювання вимог студент повинен:

- а) визначити перелік функцій, які виконує програма, і надати їх детальний опис;
- б) запропонувати інтерфейс користувача, який дозволить користувачеві отримати ці функції;
- в) скласти календарний план проекту.

Етап формулювання вимог завершується створенням календарного плану проекту та розділу пояснювальної записки «Специфікація програми».

Специфікація програми складається з послідовного опису усіх виділених функцій, та пов'язаного з ними графічного інтерфейсу користувача. У специфікації програми треба обов'язково надати ескізні зображення вікон програми.

Приклад календарного плану наведений у додатку В.

3.2 Проектування

На етапі загального проектування обирається архітектура (консольна, настільна з GUI, мережна), визначається загальна структура програми, а також формат зберігання зовнішніх даних.

Взаємодія частин та співвідношення між класами відображується за допомогою діаграм.

Проектування окремих частин програми складається зі створення об'єктної моделі частини та попереднього опису класів.

Завершується проектування створенням проектної специфікації програми. Проектна специфікація програми є другим розділом пояснювальної записки.

3.3 Кодування програми

На етапах кодування створюється похідний код програми. Він повинен бути гідним чином прокоментований і повністю задовольняти вимоги до похідного коду, наведених у додатку Б цих вказівок.

Увесь написаний похідний код має бути наведений у додатку до пояснювальної записки. У тексті основних розділів записки можна наводити лише невеликі за розміром окремі фрагменти коду.

3.4 Тестування програми

Існує два основних різновиди тестування: функціональне та модульне. Функціональне тестування перевіряє, як програма виконує свої функції в умовах коректних і некоректних дій користувача. Воно планується на етапі проектування і виконується у міру того, як кодуються окремі функції програми і перед захистом програми.

Модульне тестування робиться одночасно із кодуванням і має за мету автоматичну перевірку окремих частин програми. Виконання модульного тестування дуже бажано, але не є обов'язковим.

3.5 Пояснювальна записка

Оформлення пояснювальної записки є останнім етапом виконання курсової роботи і має надати студентові навичок документування програмного продукту. Склад та вимоги до оформлення пояснювальної записки наведено в наступних розділах.

4 СКЛАД ПОЯСНЮВАЛЬНОЇ ЗАПИСКИ

За результатами курсового проектування кожен студент індивідуально складає пояснювальну записку.

Структура пояснювальної записки до курсової роботи:

- а) титульний аркуш – 1 с. (див. додаток Г);
- б) аркуш завдання на курсове проектування та календарний план (див. додаток В);
- в) реферат – 1 с. (див. додаток Д);
- г) зміст (див. додаток З);
- д) вступ – щонайменше 1 повна сторінка;
- е) основні розділи (специфікація програми, проектування програми, інструкція користувача) – 15–30 с.;
- ж) висновки – 1 с.;
- з) перелік посилань – 5–10 найменувань (див. додаток Е);
- и) додатки.

Титульний аркуш містить інформацію про дисципліну, за якою виконувалося проектування, тему індивідуального завдання, автора пояснювальної записки. Форма титульного аркуша наведена у додатку Г.

Реферат містить:

- відомості про обсяг пояснювальної записки, кількість рисунків, таблиць, додатків, використаних джерел;
- текст реферату;
- перелік ключових слів.

Текст реферату має відображати мету курсового проектування, метод розв'язання задачі, отримані результати.

Перелік ключових слів має включати від 5 до 10 слів або сполучень із тексту записки, що більшою мірою характеризують її зміст і забезпечують можливість інформаційного пошуку. Ключові слова наводяться в називному відмінку і друкуються великими літерами у рядок через кому. Зразок оформлення реферату наведено у додатку Д

У вступі стисло викладають мету роботи й галузь застосування.

Суть пояснювальної записки викладають, розділяючи матеріал на розділи. Розділи можуть поділятися на підрозділи і пункти. Кожний пункт має містити закінчену інформацію. Обов'язковими розділами записки є:

- специфікація програми (містить опис окремих функцій програми та опис інтерфейсу користувача);
- проектна специфікація (містить опис об'єктної структури та даних);
- інструкція користувача (пояснює, як встановлювати програму та користуватися нею).

У висновках дають оцінку отриманих результатів роботи, окреслюють шляхи вдосконалення та розвитку програми, пропонують області використання результатів роботи.

5 ВИМОГИ ДО ОФОРМЛЕННЯ КУРСОВОЇ РОБОТИ

5.1 Загальні вимоги

Мова пояснювальної записки – українська, або іноземна, за погодженням керівника та завідувача кафедри.

Пояснювальна записка оформляється у вигляді друкованого документа, що відповідає вимогам стандарту ДСТУ 3008-95 на оформлення науково-технічної документації [4].

Перелік посилань оформляється згідно з ДСТУ ГОСТ 7.1:2006 [5].

Документ виконується на стандартному аркуші А4 (210x297 мм). Текст і таблиці друкуються чорним кольором.

5.2 Вимоги до тексту

Для тексту, таблиць використовується:

- шрифт Times New Roman;
- розмір шрифту 12 або 14 пт;
- інтервал абзацу – «Полуторный», після та перед рядком інтервал дорівнює 0 пт;
- інтервал абзацу – «Одинарный», шрифт Courier New для наочності подання програмного коду у записці;
- абзацний відступ дорівнює 1,25 см.;
- поля: ліве – 3 см., праве – 1,5 см, верхнє і нижнє – 2см.

Необхідно витримувати рівномірну щільність, контрастність і чіткість зображення по всьому документові. Помилки, описки і графічні неточності виправляти шляхом підчищення або зафарбування білою фарбою і нанесення на тому ж місці потрібного тексту машинним способом або від руки. Виправлення має бути чорного кольору.

Текст пояснювальної записки має бути структурований за допомогою структурних та основних розділів. Заголовки структурних розділів («РЕФЕРАТ», «ЗМІСТ», «ВСТУП», «ВИСНОВКИ», «ПЕРЕЛІК ПОСИЛАНЬ») і заголовки основних розділів («1 СПЕЦИФІКАЦІЯ ПРОГРАМИ», «2 ПРОЕКТНА СПЕЦИФІКАЦІЯ», «3 ІНСТРУКЦІЯ КОРИСТУВАЧА») треба розташовувати всередині рядка і друкувати великими літерами без крапки наприкінці, не підкреслюючи. Розділи можуть розділятися на підрозділи, підрозділи – на пункти. Підрозділи повинні мати заголовки. Пункти можуть мати заголовки. Заголовки підрозділів, пунктів треба починати з абзацного відступу і друкувати малими літерами, крім першої великої, не підкреслюючи, без крапки наприкінці. Якщо заголовок складається з двох або більше речень, їх розділяють крапкою. Відстань між основним текстом і рядком із заголовком (розділу, підрозділу, пункту) складає 2 рядки або 40 пунктів. Відстань між рядками заголовка (розділу, підрозділу, пункту) або між двома заголовками приймають таку, як в основному тексті. Приклад оформлення тексту пояснювальної записки наведено у додатку Ж.

Кожний розділ починається з нового аркуша.

Не допускається розміщувати в кінці аркуша заголовки підрозділу, пункту, якщо після цього текст займає до двох рядків.

Підрозділ, пункт не може закінчуватися рисунком, таблицею, схемою, обов'язково після них має бути щонайменше два рядки тексту.

Забороняється застосовувати напівжирний шрифт. Курсивом можуть оформлюватися елементи формули в тексті, для однозначного трактування.

Якщо в пояснювальній записці прийнята специфічна термінологія, або вживаються малопоширені скорочення, нові символи, позначення і ін., їх перелік необхідно надати в окремому списку. Але, якщо спеціальні символи, терміни, скорочення, і позначення наводяться менше трьох разів, перелік їх не складається, а їхнє тлумачення роблять у тексті при першому згадуванні.

5.3 Нумерація

Нумерація розділів, підрозділів, пунктів складається арабськими цифрами.

Після номера розділу, підрозділу, пункту, підпункту крапку не ставлять.

Номер підрозділу складається з номера розділу і порядкового номера підрозділу, що розділяються крапкою.

Наприклад:

1.2 Назва другого підрозділу 1-го розділу

Номер пункту складається з номера розділу, номера підрозділу і порядкового номера пункту, що розділяються крапкою.

Наприклад:

1.2.1 Назва першого пункту другого підрозділу 1-го розділу

Нумерація рисунків, таблиць, формул має бути за розділами. Рисунки таблиці, формули розташовуються в тексті після першої їх згадки, або після посилання на них.

Сторінки пояснювальної записки треба нумерувати арабськими цифрами, дотримуючись наскрізної нумерації для всього тексту (починаючи з титульного аркуша). Номера ставляться у правому верхньому куті, починаючи з аркуша, який розміщується після аркуша «ЗМІСТ». Всі попередні аркуші рахуються, але не нумеруються.

5.4 Перелічення

В тексті записки можуть бути використані перелічення. Перед переліченням ставлять двокрапку. Перед кожною позицією перелічення треба ставити малу літеру українського алфавіту з дужкою або, без нумерації – дефіс (це перший рівень деталізації). Для подальшої деталізації перерахування слід використовувати арабські цифри з дужкою (другий рівень деталізації).

Приклад:

- а) перша версія бібліотеки;
- б) друга версія бібліотеки:
 - 1) для персональних комп'ютерів;
 - 2) для мобільних пристроїв;
- в) третя версія бібліотеки.

5.5 Оформлення таблиць

Таблицю слід розташовувати безпосередньо після тексту, в якому вона згадується вперше або на наступній сторінці по центру аркуша. На таблицю мають бути посилення в тексті. Таблиці слід нумерувати арабськими цифрами порядковою нумерацією в межах розділу, за винятком таблиць, що наводять у додатках. Номер таблиці складають із номера розділу і порядкового номера таблиці, розділених крапкою. Таблиця може мати назву, яку поміщають над таблицею. Назва стовпців таблиці розміщується по центру комірки таблиці. Якщо рядки або графи таблиці виходять за формат сторінки, таблицю поділяють на частини, розміщуючи одну частину під іншою, або поруч, або переносячи частину таблиці на наступну сторінку. При цьому в кожній частині таблиці повторюють її головку і боковик. При розподілі таблиці на частини припускається її головку і боковик замінити відповідно номерами граф. Слово «Таблиця <номер>» вказують один раз з абзацу ліворуч над першою частиною таблиці, над іншими частинами з абзацу ліворуч пишуть: «Продовження таблиці <номер>».

Відстань між таблицею та основним текстом становить один рядок або 20пт.

Наприклад:

Текст текст текст текст текст текст текст.

(рядок)

Таблиця 4.1 – Назва таблиці

№	Назва стовпця	Назва стовпця
1	Текст текст текст текст текст текст текст	Текст текст текст
2	Текст текст текст текст текст текст текст	Текст текст текст

(рядок)

Текст текст текст текст текст текст текст.

Якщо таблиця має розрив, то це оформлюється так:

На 1-му аркуші

Таблиця 4.2 – Назва таблиці

№	Назва стовпця	Назва стовпця
1	Текст текст текст текст текст текст текст	Текст текст текст
2	Текст текст текст текст текст текст текст	Текст текст текст

На 2-му аркуші

Продовження таблиці 4.2

№	Назва стовпця	Назва стовпця
3	Текст текст текст текст текст текст текст	Текст текст текст
4	Текст текст текст текст текст текст текст	Текст текст текст

Або

На 1-му аркуші

Таблиця 4.2 – Назва таблиці

№	Назва стовпця	Назва стовпця
1	2	3
1	Текст текст текст текст текст текст текст	Текст текст текст
2	Текст текст текст текст текст текст текст	Текст текст текст

На 2-му аркуші

Продовження таблиці 4.2

1	2	3
3	Текст текст текст текст текст текст текст	Текст текст текст
4	Текст текст текст текст текст текст текст	Текст текст текст

5.6 Оформлення рисунків

Ілюстрації (креслення, рисунки, графіки, схеми, діаграми, фотознімки) слід розташовувати в записі безпосередньо після тексту, в якому вони згадуються або на наступній сторінці. Якщо ілюстрація після посилання на неї не вміщується на сторінці, її розміщують на наступній сторінці, або у додатку.

На всі ілюстрації мають бути посилання в записі. Ілюстрації повинні мати назву, яку розміщують під ілюстрацією по центру сторінки. Ілюстрація позначається словом «Рисунок <номер> – Назва рисунка», що разом із назвою ілюстрації розміщують під рисунком. За необхідності під ілюстрацією розміщують роз'яснювальні дані (підрисунковий текст).

Ілюстрації слід нумерувати арабськими цифрами порядковою нумерацією в межах розділу, за винятком ілюстрацій, що наводяться в додатках. Номер ілюстрації містить номер розділу і порядковий номер ілюстрації, що розділені крапкою.

Наприклад:

Рисунок 3.1 – Схема алгоритму

У додатках у номер рисунка входить літера додатка.

Наприклад:

Рисунок А.1 – Об’єктна модель

Якщо ілюстрація не вміщується на одній сторінці, можна переносити її на інші сторінки, при цьому назву ілюстрації розміщують на першій сторінці, пояснення на кожній сторінці, і під ними вказують: «Рисунок __, аркуш __».

Відстань між ілюстрацією та основним текстом становить один рядок або 20пт, за умови виконання вимог п.2.4 даних методичних вказівок.

Якщо роз’яснення мають розповідний характер, то посилання на рисунок пишуть так: ...на рисунку 2.2 наведено діаграму.

Наприклад.

Текст текст текст текст текст текст текст текст текст текст текст текст.

(рядок)

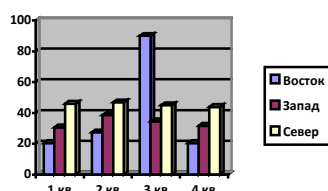


Рисунок 2.2 – Назва рисунка

(рядок)

Текст текст текст текст текст текст текст текст текст текст текст текст.

Якщо йде роз’яснення і необхідне посилання на рисунок, то в кінці абзацу або в реченні ставиться посилання на рисунок у круглих дужках, наприклад(див. рис.2.2).

Особливо необхідно приділити увагу чіткості ілюстрації, щоб читач без будь-якої напруги для очей зміг побачити окремі деталі.

5.7 Оформлення частин програмного коду

Частина програмного коду, за необхідності, може бути наведена в записці у вигляді тексту, якщо роз’яснення мають розповідний характер, або у вигляді рисунка, якщо є роз’яснення програмного механізму з посиланнями на частину коду.

Наприклад:

Наведемо програмну реалізацію властивості, що дозволяє отримати довжину списку (рекурсивний варіант):

```
public int Length
{
    get
    {
        if (next == null) return 1;
        else return next.Length + 1;
    }
}
```

Або

Приклад методу наведено на рисунку 4.3.

(рядок)

```
public void AddLast(int i)
{
    if (next == null)
        next = new DRL(i, null, this);
    else next.AddLast(i);
}
```

Рисунок 4.3 – Рекурсивна версія методу додавання (рядок)

Увесь написаний похідний код має бути наведений у додатку до пояснювальної записки. Але, якщо текст програми більш ніж 10 сторінок, допустимо навести в додатку тільки частину коду програми, але обов'язково додати до пояснювальної записки диск з усім кодом програми.

5.8 Складання переліку посилань

Перелік посилань складається згідно з ДСТУ ГОСТ 7.1:2006 [5]. Бібліографічні описи в переліку посилань наводять у порядку першого згадування в тексті.

Посилання у тексті на друковані джерела мають подаватися в квадратних дужках, наприклад: [5].

Приклад складання переліку посилань та бібліографічного опису наведено у додатку Е.

5.9 Додатки

Додатки необхідно оформляти як продовження пояснювальної записки на наступних її сторінках, розташовуючи додатки в порядку появи посилань на них у тексті пояснювальної записки. Кожний додаток повинен починатися з нової сторінки і мати заголовок, надрукований вгорі малими літерами з першої великої, симетрично щодо тексту сторінки. Посередині рядка над заголовком малими літерами з першої великої має бути надруковано слово «Додаток» і велика буква, що позначає додаток. Додатки слід позначати послідовно великими літерами українського алфавіту, за винятком букв Г, Є, З, І, Ї, Й, О, Ч, Ь. Один додаток позначається як А. Додатки повинні мати загальну з іншою частиною пояснювальної записки наскрізну нумерацію. Текст кожного додатка, за необхідності, може бути розділений на підрозділи і пункти, що нумеруються арабськими цифрами в межах кожного додатка.

6 ОРГАНІЗАЦІЯ ПРОЕКТУВАННЯ І ЗАХИСТ РОБОТИ

До захисту курсової роботи студент допускається викладачем, який керує курсовим проектуванням. Необхідними умовами допуску до захисту є:

- наявність діючої програми, яка відповідає проектному завданню;
- правильно оформлений похідний код програми;
- відповідність пояснювальної записки вимогам, викладених у попередніх розділах;
- наявність компакт-диску, що містить точну електронну копію пояснювальної записки (у форматі *.doc або *.docx) та файли коду програми.

Захист відбувається перед комісією з трьох викладачів у присутності студентів з академічної групи автора.

Склад комісії:

- голова комісії – завідувач кафедри;
- члени комісії: лектор з дисципліни та викладач, що керує курсовим проектуванням.

Студент під час доповіді повинен викласти основні результати курсової роботи. Орієнтовний час виступу – 10 хвилин. Після доповіді студент повинен відповісти на поставлені запитання. Якщо з відповідей на запитання з'ясується, що студент не є єдиним автором курсової роботи, вона відхиляється, а студент отримує нове завдання.

Курсова робота оцінюється за такими критеріями:

- якість програмного продукту та інтерфейсу користувача;
- якість похідного коду;
- виступ та відповіді на запитання;
- якість пояснювальної записки;
- ступінь відповідності термінів виконання окремих етапів календарному плану.

ПЕРЕЛІК ПОСИЛАНЬ

1. Троелсен, Э. Язык программирования С# 5.0 и платформа .NET 4.5, 6-е изд.]: пер. с англ./ Э. Троелсен. – М.: Вильямс, 2013. – 1312 с.

2. Бондарев, В.М. Объектно-ориентированное программирование на С#: учеб. пособ. /В.М. Бондарев. – Х. : Компания СМИТ, 2009. – 224 с.

3. Юркин А.Г. Задачник по программированию / А.Г. Юркин. – СПб.: Питер, 2002. – 192 с.

4. ДСТУ 3008-95 «Документація. Звіти в сфері науки і техніки. Структура і правила оформлення» – Державний стандарт України.

5. ДСТУ ГОСТ 7.1:2006 «Система стандартів з інформації, бібліотечної та видавничої справи. Бібліографічний запис. Бібліографічний опис. Загальні вимоги та правила складання».

Додаток А

Приклади тем курсових робіт

Приклади тем взяті з книги [3] та доопрацьовані.

Картотека Інтерполу. Дані по кожному зареєстрованому злочинцю: прізвище, ім'я, кличка, зріст, колір волосся і очей, особливі прикмети, громадянство, місце та дата народження, останнє місце проживання, знання мов, злочинна професія, остання справа і т.д. Злочинні та мафіозні угруповання (дані про співучасників). Вибірка з будь-якої підмножини ознак. Перенесення тих, що виправилися, в архів; видалення – тільки після смерті.

Бюро знайомств. База потенційних женихів і наречених: стать, реєстраційний номер, дата реєстрації, відомості про себе, вимоги до партнера. Вибір підмножини підходящих кандидатур, підготовка зустрічей (формування запрошення для знайомства). Перенесення в архів пар, які вирішили свої сімейні проблеми, видалення клієнтів, які відмовилися від послуг.

Біржа праці. База безробітних: анкетні дані, професія, освіта, місце і посада останнього працевлаштування, причина звільнення, сімейний стан, житлові умови, контактні координати, вимоги до майбутньої роботи. База вакансій: фірма, посада, умови праці і оплати, житлові умови, вимоги до фахівця. Пошук та реєстрація варіантів з того чи іншого боку; формування оголошень для друку, видалення в архів після працевлаштування, повне видалення при відмові від послуг.

Записна книжка. Анкетні дані, адреси, телефони, місце роботи або навчання, посада знайомих, колег і родичів, характер знайомства, ділові якості тощо. Автоматичне формування привітання з днем народження (за поточною датою). Впорядкування за алфавітом і за датою останнього коригування. Пошук за довільним шаблоном.

Каса аерофлоту. Розклад: номер рейсу, маршрут, пункти проміжної посадки, час відправлення, дні польоту. Кількість вільних місць на кожному рейсі. Вибір найближчого рейсу до заданого пункту (за наявності вільних місць), оформлення заданого числа квитків із узгодженням з пасажиром (зі зменшенням числа вільних місць), оформлення посадкової відомості.

Довідник споживача (служба побуту). База підприємств побутового обслуговування міста: назва, розряд, адреса і телефони, спеціалізація, перелік послуг, що надаються, форма власності, години і дні роботи. Пошук підприємств за заданою послугою та іншими ознаками. Вивід у файл списку обраних підприємств з контактами.

Довідник покупця. База торгових точок міста: назва, адреса і телефони, спеціалізація, форма власності, час роботи. Вибір крамниць за довільним шаблоном. Вивід у файл списку обраних підприємств з контактами.

Магазин з одним продавцем. Комп'ютер замість касового апарату. База наявності товарів: найменування, одиниця виміру, ціна одиниці, кількість, дата останнього завезення. Реєстрація надходження товару (як старих, так і нових найменувань). Оформлення покупки: виписка чека, коректування бази. Проблема уцінки і списання. Інвентаризація залишків товару з обчисленням сумарної вартості.

Відділ кадрів. База даних про співробітників фірми: паспортні дані, освіта, спеціальність, підрозділ, посада, оклад, дати надходження у фірму і останнього призначення та ін. Вибір за довільним шаблоном. Скорочення штатів: вибір для звільнення осіб пенсійного та передпенсійного віку. Підготовка наказів.

Генеалогічне дерево. Паспортні дані членів деякого родового клану; посилання на дітей (або на батьків). Пошук усіх нащадків або всіх предків для вказаної особи. Ієрархічне відображення генеалогічного дерева обраної людини.

Склад. База товарів, що зберігаються на складі: найменування, одиниця виміру, ціна одиниці, кількість, дата останнього завезення. Реєстрація надходження товару (формування прибуткової накладної) і відвантаження (видаткова накладна). Вивід інвентарної відомості.

Каса автовокзалу. Розклад автобусів: номер рейсу, кінцевий і проміжні пункти, час відправлення. Кількість вільних місць на кожному рейсі. Вибір найближчого рейсу до заданого пункту (за наявності вільних місць), оформлення квитків, оформлення посадкової відомості. Попередній продаж, повернення квитків.

Адміністратор готелю. Список номерів: клас, число місць. Список гостей: паспортні дані, дати приїзду і від'їзду, номер. Поселення гостей: вибір відповідного номера (за наявності вільних місць), реєстрація, оформлення квитанції. Від'їзд: вибір всіх заїжджих, від'їжджаючих сьогодні, звільнення місця або оформлення затримки з випискою додаткової квитанції. Можливість дострокового від'їзду з перерахунком. Пошук гостя за довільною ознакою.

Довідник меломана. База груп і виконавців; база пісень; база дисків з переліком пісень (у вигляді посилань). Вибір всіх пісень заданої групи; всіх дисків, де зустрічається задана пісня.

Щоденник. База запланованих заходів – дата, час і тривалість, місце проведення. Автоматичне нагадування найближчої справи: за поточною датою і часом; видалення вчорашніх справ або перенесення на майбутнє. Аналіз «накладок» – перетинань планованих справ. Перегляд справ на завтра, післязавтра і т.д.

Термінологія. База визначень будь-якої науки: вводиться термін, його тлумачення (визначення), посилання на терміни, що використовуються, (за типом гіперпосилань). Можливість перегляду всього ланцюжка від заданого терміну до первинних понять.

Шеф – кухар. База рецептур страв: розкладка, рецепт приготування. База продуктів на складі: найменування, ціна, кількість, термін придатності. Формування меню на день (на задане число персон). Пропозиція страви дня з урахуванням запасів продуктів. Перевірка достатності запасів; формування видаткової накладної на склад, коригування запасів.

Довідник лікаря. База хвороб: назва, симптоми, процедури, перелік рекомендованих ліків із зазначенням необхідної кількості. База медикаментів на складі: назва, кількість, взаємозамінність. Формування рецепта після огляду хворого, перевірка наявності ліків, коригування запасів.

Зарахування абітурієнтів. База абітурієнтів: анкетні дані, сукупність оцінок на вступних іспитах, готовність вчитися на договірній основі. Вибір для зарахування заданої кількості абітурієнтів; формування для співбесіди списку тих, хто набрав граничний прохідний бал, але не може платити за освіту.

Обмін житла. База пропозицій з обміну: район, площа, планування тощо; вимоги до варіантів обміну. Реєстрація клієнтів, вибір відповідних варіантів, видалення при обміні або відмові. Можливість з'їзду або роз'їзду, в тому числі «кілька на кілька»; «можливі варіанти».

Довідник абітурієнта. База вузів: найменування, адреса, перелік спеціальностей, конкурс минулого року за кожною спеціальністю (денною, вечірньою, заочною формами), розмір оплати при договірному навчанні. Вибір за різними критеріями: все щодо обраного вузу; все щодо обраної спеціальності, пошук мінімального конкурсу з даної спеціальності та ін.

Довідник поштової індексації. Країна, область (край), район, населений пункт, поштовий індекс. Пошук за будь-якою сукупністю полів, ієрархічних зв'язків між полями.

Банк. Відомості про вкладників банку: номер особового рахунку, категорія вкладу, паспортні дані, поточна сума вкладу, дата останньої

операції. Операції прийому і видачі будь-якої суми, автоматичне нарахування відсотків.

Ломбард. База збережених товарів і нерухомості: анкетні дані клієнта, найменування товару, оцінна вартість; сума, видана під заставу, дата здачі, термін зберігання. Операції прийому товару, повернення, продажу після закінчення терміну зберігання.

Довідник селекціонера. Найменування сорту культури, автор, батьківські сорти, урожайність, характеристики плодів, морозостійкість, стійкість до шкідників і хвороб, наявність в тому чи іншому селекційному фонді. Вибір сортів, що володіють заданими властивостями.

Довідник працівника ДАІ. Марка, колір, заводський і бортовий номер, дата випуску, особливості конструкції і забарвлення, дата останнього техогляду транспортного засобу (автомобіля, мотоцикла, причепа тощо), паспортні дані власника. Вибір транспортних засобів за довільним шаблоном. Формування запрошень на техогляд відповідно до терміну.

Довідник власника відеотеки. База відеофайлів: назва фільму, студія, жанр, рік випуску, режисер, виконавці головних ролей, короткий зміст, суб'єктивна оцінка фільму, розташування, розмір. Перевірка наявності фільму у відеотеці. Підбір фільму для перегляду за рядом критеріїв, підбір. Складання вибірки ряду фільмів за часом відтворення, розміру зовнішнього носія, жанру і акторського і режисерського складу.

Біржа товарів. База продавців: найменування товару, обсяг партії при оптовому продажі, ціна, умови продажу-відвантаження, форма оплати, контактна адреса або телефон, примітка (наприклад, «посередників прошу не турбуватися»). База покупців: найменування товару, обсяг покупки, прийнятна ціна та форма оплати, контактна адреса або телефон, примітка. Пошук і реєстрація варіантів з того чи іншого боку; формування оголошень для друку, видалення в архів після купівлі-продажу (можливо, один з клієнтів залишається незадоволеним), повне видалення при відмові від послуг.

Довідник фаната. База спортсменів: анкетні та антропологічні дані, громадянство, походження, вид спорту, клуб або команда, дані про особистий рекорд або перемоги та ін. Вибір за довільною ознакою. Пошук рекордсмена в заданому виді спорту.

Довідник радіолюбителя. Бази паспортних даних транзисторів, діодів, тиристорів тощо: марка, характеристики, гранично допустимі умови експлуатації, ціна, облік взаємозамінності та ін. Підбір по заданих вимогах.

Довідник комерційних банків. Найменування, адреса, статус (форма власності), умови зберігання коштів на особовому рахунку (річні відсотки на різних видах вкладів). Вибір банку з найкращими умовами для заданого типу вкладу.

Довідник начальника в'язниці. Анкетні дані ув'язнених, стаття, термін, дата взяття під варту, місце в тюремній ієрархії, камера, відомості про родичів, особливості характеру. Формування статистичних зведень про склад, вибір за довільною ознакою.

Довідник командира. Список підлеглих військовослужбовців: анкетні дані, адреса батьків, громадянська професія, освіта, звання та дата його отримання, посада, підрозділ, форма служби (термінова, кадрова, контрактна та ін.), період служби, особливості характеру та ставлення до служби. Формування списків: заданого підрозділу, офіцерського складу, новобранців та ін.

Ріелтерська контора (купівля-продаж житла). База пропозицій: район та адреса, характеристика будинку та квартири, запитувана вартість, координати заявника. База попиту: вимоги покупця до житла (можливо декілька варіантів, допустимі діапазони), фінансові можливості, координати заявника. Підбір варіантів для того чи іншого боку, автоматизований пошук взаємоприйнятних варіантів.

Черга на житло. Список черговиків на отримання та поліпшення житла: дата надходження у фірму, дата подачі заяви, склад сім'ї, житлові умови на дату подачі заяви, пільги на додаткову площу, позачергове та першочергове поліпшення, побажання на район та ін. Реалізація розподілу одержуваного муніципального житла: задоволення відповідних черговиків, переселення наступних в квартири, що звільнилися. Додавання заявників та видалення вибулих та задоволених.

Автосалон. База нових і старих вітчизняних та іноземних автомобілів: марка, рік випуску, технічні характеристики, особливості виконання, технічний стан, запитувана ціна. База покупців: контактні координати, вимоги до марки, технічних характеристик та технічного стану, фінансові можливості. Автоматизація підбору варіантів для покупця, формування заявки для постачальників та перевізників.

Довідник туриста. Турагенства та запропоновані послуги: країна, місто (або маршрут круїзу), умови проживання та проїзду, екскурсійне обслуговування, сервіс приймаючої сторони, вартість путівки. Підбір туру за списком міст, які хочеться відвідати, за іншими критеріями.

Купівля продаж. База оголошень (радіо-, газетних, у рухомому рядку) по всіх рубриках. Пошук для будь-якого попиту або пропозиції, включаючи контекстний пошук в МЕМО-полях (наприклад, «репетиторство & математика»).

Крилаті вислови. Довідник прислів'їв, приказок, афоризмів, каламбурів, інших словесних курйозів. Класифікація за автором та джерелами, пошук за темами та ключовими словами. Підготовка обраних афоризмів для друку.

Каталог запчастин автомобіля. В автомобілі налічується декілька тисяч деталей, деякі використовуються в різних марках. Таблиці: країна, фірма-виробник, марка автомобіля, агрегат, вузол, деталь. Облік взаємозамінності. Користувачі: працівники автосервісу, магазинів запчастин; постачальники-оптовики.

Каталог радіодеталей (довідник радіомайстрів). Моделі побутової аудіо- та відеотехніки; для кожної моделі – каталог радіодеталей, використаних в ній (резистори, конденсатори, котушки індуктивності, чіпи та ін.). Багато деталей використовуються в різних моделях; деякі деталі односторонньо (наприклад, замість резистора 10К 0,125Вт можна поставити 10К 0,25Вт, але не навпаки) або двосторонньо (транзистори з близькими характеристиками) замінні. Облік наявності деталей на складі, пошук відповідних деталей для ремонту.

Довідник нумізмата. Монети: країна, номінал, рік випуску, метал або сплав, кількість випущених монет, особливості. Колекціонери: країна, ім'я, контактні координати, наявність рідкісних монет у колекції. Власна колекція.

Довідник філателіста. Марки: країна, номінальна вартість, рік випуску, тираж, особливості. Філателісти: країна, ім'я, контактні координати, наявність рідкісних марок у колекції. Власна колекція.

Довідник любителя живопису. Художники з анкетними даними та стилями. Картини з посиланням на художників, датою створення, жанром. Колекціонери та музеї: наявність оригіналів картин і копій. Аукціони та комісійні магазини: дата проведення, список виставлених шедеврів та ціни на них. Власна колекція.

Телепрограма. Програма телепередач декількох телекомпаній (на тиждень, по днях, годинах). Різні жанри телепередач: новини, спорт, художні фільми, серіали та ін. Вибір сукупної програми за певним запитом. Збереження списку обраних передач.

Довідник астронома. Для кожної з зареєстрованих зірок відомі: назва, сузір'я, видима зоряна величина, відстань, координати на небосхилі. Пошук зірок заданого сузір'я, найяскравіших зірок, видимих зірок і сузір'їв у заданій точці земної кулі в заданий час. Вибір часу спостереження за обраною зіркою.

Довідник географа. Міста (географічні координати, чисельність населення), регіони (вид, приналежність країні, столиця, чисельність населення), країни (площа, чисельність населення, форма державного правління, столиця), материки. Пошук за певними критеріями, показ на карті розташування, населеність материків та ін.

Довідник гідролога. Річки світу: довжина (від джерела до впадіння), куди впадає (в іншу річку, море або озеро), річний стік, площа басейну (без великих приток). Обчислення річного стоку та площі басейну для заданої великої річки (разом з її притоками), моря або озера.

Термінологічний довідник астронома. Існуючі небесні тіла: назва, визначення, вид, розмір, відстань від Землі, фотографія. Ієрархічні зв'язки між термінами. Підготовка до друку обраних термінів.

Особиста бібліотека. Картотека домашньої бібліотеки: вихідні дані книги (автори, назва, видавництво та ін.), розділ бібліотеки (спеціальна література, хобі, домашнє господарство, белетристика та ін.), походження та наявність книги в даний момент, суб'єктивна оцінка книги. Вибір книг за довільним запитом; інвентаризація бібліотеки.

Додаток Б

Узгодження щодо кодування програм

Курсову роботу слід виконувати мовою C# або java script. Нижче викладені вимоги до вихідного коду програми мовою C #.

1 Структура файлів

1.1 Структура проекту

Кожен проект має розташовуватися в окремому підкаталозі каталогу рішення. Ніколи не кладіть файли проекту та файли рішення в один та той самий каталог.

Використовуйте підкаталоги для організації вихідного коду проекту.

Простори імен мають відповідати структурі каталогів.

1.2 Структура файла вихідного коду

Файл вихідного коду повинен мати таку структуру (порядок елементів також має значення):

- а) using – декларації;
- б) Декларації простору імен;
- в) За необхідності перелічіть допоміжні структури, потрібні для даного класу.
- г) Оголошення класу. Рекомендується оголошувати один клас в одному файлі.

1.3 Структура класів

Класи та структури мають оголошуватися у такому порядку:

- а) Поля;
- б) Конструктори;
- в) Властивості (спочатку public, потім protected, потім private);
- г) Методи (спочатку public, потім protected, потім private);
- д) Делегати та події.

2 Форматування коду

Код має бути відформатований відповідно до таких правил:

- а) Параметр «Tabs» має бути встановлений у положення «Keep spaces»;
- б) Розмір відступу має бути 4 пробіли;
- в) Програмні конструкції вибору та повторення мають бути відформатовані, як у таких прикладах:

```
if (file.Exists(fileName))
{
    file.Open(fileName);
}
```

```
for (int i = 0; i < MAX_ELEMENTS; ++i)
{
    array[i] = i * MULTIPLY_FACTOR;
}
```

```
switch (workMode)
{
    case WorkMode.Add:
        DoAdd();
        break;
    case WorkMode.Update:
        DoUpdate();
        break;
}
```

г) Порожні рядки використовуються для поліпшення сприйняття коду. Зазвичай вони використовуються для поділу незалежних секцій коду. Обов'язково розміщуйте порожній рядок:

1) Після останнього оголошення «using» перед оголошенням простору імен;

2) Між методами;

3) Між оголошенням локальних змінних методу та першою інструкцією методу;

4) Перед багаторядковим та однорядковим коментарем за винятком коментаря після фігурної дужки, що відкриває блок;

5) Перед логічно ізольованою частиною коду методу.

д) Пробіли також слід використовувати для поліпшення сприйняття коду. Необхідно вставляти пробіли:

1) Перед круглими дужками, наступними за ключовим словом:

```
if (condition)
```

2) Після коми в списку аргументів:

```
int result = Calculate(argumentOne, argumentTwo);
```

3) Між бінарним оператором та його операндами:

```
int result = argumentOne + argumentTwo;
```

4) Між частинами інструкції "for":

```
for (int i = 0; i < MAX_ELEMENTS; ++i)
```

е) Аргументи унарних операторів ніколи не відокремлюються пробілами:

```
i++;  
--counter;
```

ж) Всі довгі рядки мають бути згорнуті. Довжина рядка не має перевищувати 80 символів. При згортанні довгих рядків намагайтеся дотримуватися таких правил:

- 1) Переносить рядок після коми;
- 2) Переносить рядок перед оператором;
- 3) Віддавайте перевагу «логічному» розбиттю рядка, а не «фізичному»;
- 4) Робіть додатковий відступ перед «згорнутою» частиною рядка:

```
int i;  
do  
{  
  
    // Here comes a long line.  
    bool isSucceeded = a + b + c + d + e - h * MAX_FACTOR /  
DEFAULT_SCALE  
    - correctionFactor;  
}
```

з) Довгий список параметрів методу також має бути згорнутий, так само як список аргументів у інструкції виклику методу:

```
// The method declaration.  
public float DoSomethingFromManyArguments(  
    int argumentOne,  
    int argumentTwo,  
    decimal argumentThree,  
    string argumentFour)  
{  
    // Method body goes here.  
}  
  
// The calling code.  
float result = DoSomethingFromManyArguments(  
    1,  
    1,  
    0.5m,  
    "Some text");
```

и) Не оголошуйте декілька змінних в одному рядку, використовуйте окремий рядок для кожної змінної:

```
int    customerID;  
string customerName;  
float  salary;
```

3 Угоди про найменування

Не використовуйте угорську або іншу префіксну нотацію за винятком випадків, обумовлених далі.

Імена локальних змінних та аргументів методів мають записуватися в нотації «верблюду»:

```
string customerName;  
  
private int DoSomething(int firstArgument, float secondArgument);
```

Імена закритих полів класу мають наслідувати нотації «верблюду»:

```
private string customerName;
```

Імена відкритих полів класу мають наслідувати нотації «Паскаль»:

```
public int CustomerID;
```

Імена властивостей та методів класу мають наслідувати нотації «Паскаль»:

```
public bool ValidateAmount();  
public bool FileExists  
{  
    get;  
}
```

Імена класів, структур та нумераторів мають наслідувати нотації «Паскаль»:

```
public class CustomerAccount  
{  
};  
  
private enum WorkMode  
{  
    Add,  
    Update;  
}
```

Імена методів мають наслідувати шаблоном «дієслово» + «іменник» – наприклад, «UpdateAccount»;

Використовуйте однину, а не множину в іменах перелічення. Іншими словами, «WorkMode» – це правильне ім'я, а «WorkModes» – ні;

Використовуйте значущі імена навіть для закритих методів, властивостей, типів та ін. Уникайте використовувати короткі імена, такі як «a», «b», «n» за винятком загальноприйнятих імен для змінних циклу «i» і «j».

4 Загальні принципи розробки

Культура кодування:

- а) Дублювання коду в програмі суворо забороняється.
- б) Обов'язкова заміна числових і рядкових літералів символічними константами. Виняток можна зробити для самоочевидних констант, таких як 0 та 1. Давайте символічним константам значущі імена;
- в) Рекомендується об'єднувати пов'язані між собою константи цілих типів у нумератори, а не цілих типів – в абстрактні класи з відкритими статичними константними членами;
- г) Уникайте довгих та складних методів, розбивайте їх на декілька коротких. При модифікації коду максимально використовуйте рефакторинг;
- д) Завжди звертайте увагу на попередження компілятора. Подбайте, щоб їх не було.

5 Угоди про коментарі

Обов'язкові коментарі до класів та до інших призначених для користувача типів та до всіх відкритих членів класів.

Коментарі всередині коду методів не обов'язкові. Код повинен легко читатися та без коментарів.

Коментар не повинен перефразовувати те, що написано в кодї. Намагайтеся давати змістовні пояснення. Наприклад, коментар “Increment i by one” до коду “i++;” це погана практика, а коментар “Update the number of customer accounts processed” – добра.

Текст коментаря відокремлюйте від слешів одним пропуском. Перша буква речення має бути великою, в кінці речення має стояти крапка:

```
// This is a comment. This is yet another comment sentence.
```

Коментарі мають підкорятися загальним правилам згортання рядків.

Не використовуйте коментарі в стилі C: /* ... */

6 Специфіка Windows Forms

Дотримуйтесь Microsoft User Interface Guidelines, якщо до вас не висувають інших вимог.

Якщо два керуючих елементи служать одній меті (наприклад, поле введення для імені користувача та позначка перед ним), дозволяється використовувати тип елемента в закінченні імені. Наприклад, «customerNameText» та «customerNameLabel».

Слід прибрати всю бізнес-логіку з форм та керуючих елементів та розташовувати її в класах моделі. В обробники подій елементів розміщуйте тільки виклики методів.

Використовуйте нотацію «верблюду» для найменування екземплярів керуючих елементів на формах та призначених для користувача керуючих елементах, наприклад, «optionsHeading».

Використовуйте нотацію «Паскаль» для найменування класів форм і керуючих елементів, наприклад, «ProductList».

Прагніть до повторного використання UI та вихідного коду. Діліть ваш інтерфейс на елементи; сміливо наслідуйте форми та керуючі елементи.

Джерело інформації:

<http://msdn.microsoft.com/library/default.asp?url=/library/en-us/cpgenrefer/html/cpconnetframeworkdesignguidelines.asp>

Додаток В

Аркуш завдання та календарний план

ХАРКІВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ РАДІОЕЛЕКТРОНІКИ

Кафедра: *Програмної інженерії*

Дисципліна: *Об'єктно-орієнтоване програмування*

Напрямок: *Програмна інженерія*

Курс 1. Група III-15-X. Семестр 2.

ЗАВДАННЯ *на курсовий проект студента*

(Прізвище, Ім'я, По батькові)

1 Тема проекту:

2 Термін здачі студентом закінченого проекту: **“31” – травня – 2016 р.**

3 Вихідні дані до проекту:

Специфікація програми, методичні вказівки до виконання курсової роботи

4 Зміст розрахунково-пояснювальної записки:

Вступ, специфікація програми, проектна специфікація, інструкція користувача, висновки

5 Перелік графічного матеріалу:

Схема об'єктної моделі, алгоритми, приклади екранних форм

КАЛЕНДАРНИЙ ПЛАН

<i>№</i>	<i>Назва етапу</i>	<i>Термін виконання</i>
1	Видача теми, узгодження і затвердження теми	1-03-2016 р.
2	Формулювання вимог до програми	__-__-2016 – __-__-2016 р.
3	Розробка підсистеми зберігання та пошуку книг.	__-__-2016 – __-__-2016 р.
4	Розробка підсистеми зберігання та пошуку читачів.	__-__-2016 – __-__-2016 р.
5	Розробка функцій видачі та повернення книг	__-__-2016 – __-__-2016 р.
6	Розробка функцій зберігання та завантаження даних	__-__-2016 – __-__-2016 р.
7	Тестування і доопрацювання розробленої програмної системи.	__-__-2016 – 29-04-2016 р.
8	Оформлення пояснювальної записки, додатків, графічного матеріалу	__-__-2016 – 31-05-2016 р.
9	Захист	__-__-2016 р.

Студент _____

Керівник _____

(Прізвище, ім'я, по батькові)

« 1 » березня _____ 2016 р.

Додаток Г

Зразок титульного аркуша

Міністерство освіти і науки України
Харківський національний університет радіоелектроніки

Кафедра Програмної інженерії

КУРСОВА РОБОТА

ПОЯСНЮВАЛЬНА ЗАПИСКА

з дисципліни «Об'єктно -орієнтоване програмування»

<НАЗВА>

Керівник , <посада>

<прізвище, ініціали >

Студент гр. <шифр групи>

<прізвище, ініціали >

Комісія:

проф. _____ Дудар З.В.

<посада> _____<прізвище, ініціали >

<посада> _____<прізвище, ініціали >

Харків 2016

Додаток Д

Зразок оформлення реферату

РЕФЕРАТ

Пояснювальна записка до курсової роботи: 44 с., 10 рис., 3 табл., 2 додатки, 7 джерел.

Метою роботи є розробка програми «Довідник покупця» на засадах об'єктно-орієнтованого програмування.

Методи розробки базуються на використанні середовища розробки Microsoft Visual Studio 2013, Windows Forms, платформи .NET Framework 4.5, мови програмування C#.

В результаті отримана програма під назвою «Довідник покупця», яка дозволяє зберігати список магазинів, характеристики кожного магазину, такі як: назва, спеціалізація, адреса, телефон, час роботи, вид власності. Є можливість утворювати нові списки магазинів, додавати, видаляти та редагувати магазини. Також, є функція формування звіту у Microsoft Excel.

ПРОГРАМА, ОБ'ЄКТНО-ОРІЄНТОВАНЕ ПРОГРАМУВАННЯ, КЛАС, МОВА ПРОГРАМУВАННЯ C#, ПЛАТФОРМА .NET, СКЛАД, НАКЛАДНА, ВІДОМІСТЬ, РЕГІСТР.

Додаток Е

Приклад оформлення переліку посилань

ПЕРЕЛІК ПОСИЛАНЬ

1. Андреев, В. В. Как организовать делопроизводство на предприятии / В. В. Андреев. – М. : ИНФРА-М, 1997. – 94 с.
2. Белов, А. В. Финансы и кредит: учеб. / А. В. Белов, В. Н. Николаев ; КНУ им. Т. Г. Шевченко. – К. : Университет, 2004. – 215 с.
3. Агафонова, Н. Н. Гражданское право: учеб. пособие / Н. Н. Агафонова, Т. В. Богачева, Л. И. Глушкова ; под общ. ред. А. Г. Калпина ; МОН Украины. – 2-е изд., перераб. и доп. – Х. : Фактор, 2000. – 542 с.
4. Елементи інформатики [Текст]: довідник / В. С. Височанський, А. І. Кардаш, В. С. Костєв, В. В. Черняхівський. – К. :Наук. думка, 2003. – 192с.
5. Microsoft Developer Network. Библиотека MSDN. Разработка на .NET [Электронный ресурс] / MSDN – сеть разработчиков Microsoft. – Режим доступа : www/URL:https://msdn.microsoft.com/ru-ru/ – 25.05.2015 г. – Загл. с экрана.
6. ГОСТ Р 517721-2001. Аппаратура радиоэлектронная бытовая. Входные и выходные параметры и типы соединений. Технические требования. – Введ. 2002-01-01. – М. : Изд-во стандартов, 2001. – 27 с.

Додаток Ж

Зразок оформлення тексту пояснювальної записки

1 НАЗВА ПЕРШОГО РОЗДІЛУ

1.1 Назва першого підрозділу першого розділу

(рядок)

(рядок)

Текст текст текст текст текст текст текст текст текст текст текст текст
текст текст текст текст текст текст текст [1].

Текст текст текст текст текст текст текст текст текст кст текст текст текст текст.

(рядок)

(рядок)

1.2 Назва другого підрозділу першого розділу

(рядок)

(рядок)

Текст текст текст текст текст текст текст текст текст текст текст текст
текст текст текст текст.

(рядок)

(рядок)

1.2.1 Назва пункту

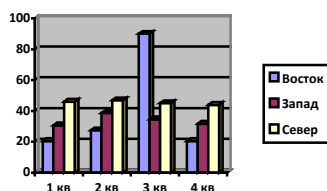
(рядок)

(рядок)

Текст текст текст текст текст. Текст текст текст текст текст текст текст
текст текст текст текст текст.

На рисунку 1.1 зображено ...

(рядок)



(рядок)

Рисунок 1.1 – Назва рисунка

(рядок)

Текст текст текст текст текст текст текст текст текст текст текст текст текст текст текст текст текст.

Текст текст текст текст текст текст текст текст текст текст текст текст текст текст текст текст текст. У таблиці 1.1 наведено...

(рядок)

Таблиця 1.1 – Назва таблиці

№	Назва стовпця	Назва стовпця
1	Текст текст текст текст текст текст текст	Текст текст текст
2	Текст текст текст текст текст текст текст	Текст текст текст

(рядок)

В формулі 2.1текст текст текст текст текст текст текст текст.

(рядок)

$$\begin{aligned}A+B &= C, \\ A-B &= C\end{aligned}\tag{2.1}$$

(рядок)

.....

Розглянемо метод на рисунку 3.1.

(рядок)

```
public void PrintRev2()  
{  
    DRL cur=this;  
    while (cur.next!=null) cur=cur.next;  
    while (cur.prev != null)  
    {  
        Console.Write(cur.info + " ");  
        cur = cur.prev;  
    }  
    Console.WriteLine(cur.info);  
}
```

(рядок)

Рисунок 3.1 – Назва першого рисунка третього розділу

(рядок)

Текст текст текст текст текст текст текст текст текст текст текст текст текст текст текст текст текст.

....

Додаток И

Приклад складання змісту

ЗМІСТ

Перелік умовних скорочень	6
Вступ	7
1 Специфікація програми.....	8
1.1 Назва підрозділу.....	8
1.2 Назва підрозділу.....	10
2 Проектна специфікація.	12
2.1 Назва підрозділу.....	12
2.1.1 Назва пункту.....	13
2.1.2 Назва пункту.....	14
2.2 Назва підрозділу.....	16
2.3 Назва підрозділу.....	18
3 Інструкція користувача	22
Висновки	26
Перелік посилань.....	27
Додаток А – Код програми.	28

Навчальне видання

МЕТОДИЧНІ ВКАЗІВКИ
до виконання курсової роботи з дисципліни
ОБ'ЄКТНО-ОРІЄНТОВАНЕ ПРОГРАМУВАННЯ

для студентів студентів усіх форм навчання
напряму підготовки 6.050103 – «Програмна інженерія»,
спеціальність 121 – «Інженерія програмного забезпечення»

Упорядники: БОНДАРЄВ Володимир Михайлович
ЧЕРЕПАНОВА Юлія Юріївна

Відповідальний випусковий З.В. Дудар

Редактор Б.П. Косіковська

Комп'ютерна верстка Г.М. Голоднікова

План 2016 (друге півріччя), поз. 26.

Підп. до друку 05.07.16. Формат 60x 84¹/₁₆. Спосіб друку – ризографія.

Умов. друк. арк. 2,3. Облік. вид. арк. 2,1. Тираж 25 прим.

Зам. № 1-26. Ціна договірна.

ХНУРЕ. Україна. 61166 Харків, просп. Науки, 14

Віддруковано в редакційно-видавничому відділі ХНУРЕ
Харків, 61166, просп. Науки, 14

