МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

ХАРКІВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ РАДІОЕЛЕКТРОНІКИ

**МЕТОДИЧНІ ВКАЗІВКИ**

до виконання курсової роботи з дисципліни

"ОБ’ЄКТНО-ОРІЄНТОВАНЕ ПРОГРАМУВАННЯ"

Харків 2022

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

ХАРКІВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ РАДІОЕЛЕКТРОНІКИ

МЕТОДИЧНІ ВКАЗІВКИ

до виконання курсової роботи з дисципліни

"ОБ’ЄКТНО-ОРІЄНТОВАНЕ ПРОГРАМУВАННЯ"

для студентів першого (бакалаврського) рівня вищої освіти усіх форм навчання спеціальності

121 – «Інженерія програмного забезпечення» освітньо-професійна програма «Програмна інженерія»

Затверджено кафедрою ПІ

Протокол № 11 від 08.02.2022 р.

Харків 2022

Методичні вказівки до виконання курсової роботи з дисципліни «Об’єктно-орієнтоване програмування» для студентів першого (бакалаврського) рівня вищої освіти усіх форм навчання спеціальності 121 «Інженерія програмного забезпечення», освітньо-професійна програма «Програмна інженерія» / Упоряд.: В.М. Бондарєв, Ю.Ю. Черепанова – Харків: ХНУРЕ, 2022. – 43 с.

# ЗМІСТ

[ВСТУП 4](#_Toc102572249)

[1. МЕТА КУРСОВОГО ПРОЕКТУВАННЯ 5](#_Toc102572250)

[2. ТЕМАТИКА КУРСОВИХ РОБІТ 6](#_Toc102572251)

[3. ПОРЯДОК ВИКОНАННЯ КУРСОВОЇ РОБОТИ 7](#_Toc102572252)

[3.1 Формулювання мети роботи 7](#_Toc102572253)

[3.2 Опис вимог до програми 7](#_Toc102572254)

[3.3 Проектування 8](#_Toc102572255)

[3.4 Кодування програми 8](#_Toc102572256)

[3.5 Тестування програми 8](#_Toc102572257)

[3.6 Завершення пояснювальної записки 8](#_Toc102572258)

[4. ВИМОГИ ДО ОФОРМЛЕННЯ ЗАПИСКИ 10](#_Toc102572259)

[4.1 Склад записки 10](#_Toc102572261)

[4.2 Загальні вимоги 10](#_Toc102572262)

[4.3 Вимоги до тексту 11](#_Toc102572263)

[4.4 Нумерація 11](#_Toc102572264)

[4.5 Переліки 12](#_Toc102572265)

[4.6 Оформлення рисунків 12](#_Toc102572266)

[4.7 Оформлення програмного коду 13](#_Toc102572267)

[4.8 Складання переліку посилань 13](#_Toc102572268)

[4.9 Додатки 13](#_Toc102572269)

[5. ЗАХИСТ РОБОТИ 15](#_Toc102572270)

[ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ 17](#_Toc102572271)

[ДОДАТКИ 18](#_Toc102572272)

# ВСТУП

Метою дисципліни "Об’єктно-орієнтоване програмування" є вивчення основних принципів ООП, та оволодіння навичками розробки об’єктних програм. Дисципліна повинна зосередити студента на найбільш важливих рисах програмування, що не залежать від таких обставин, як тип процесора або операційна система. Саме об’єктно-орієнтоване програмування надає таку можливість, бо воно абстрагується навіть від алгоритмічної мови.

За результатом вивчення дисципліни студенти повинні:

а) знати загальні принципи ООП, особливості об’єктної моделі мови С#, 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 застосувань;

в) володіти методами та технологіями об'єктно-орієнтованого

програмування.

Працюючи над курсовою роботою, студент повинен пройти усі фази розробки програмного забезпечення і в результаті створити закінчений програмний продукт.

# МЕТА КУРСОВОГО ПРОЕКТУВАННЯ

Метою курсового проектування є закріплення знань, набутих при вивченні дисципліни „Об’єктно-орієнтоване програмування”, а головне, придбання навичок у розробці програмних продуктів.

Студент повинен надбати такі властивості як впевненість у власних знаннях, вміння відстоювати власну думку, вміння виступати перед аудиторією і проявити їх під час захисту.

# ТЕМАТИКА КУРСОВИХ РОБІТ

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

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 змінювати не можна.

Незалежно від теми програма повинна відповідати таким загальним вимогам.

*Стійкість програми.* Програма не повинна втрачати працездатності за жодних, навіть некоректних, дій користувача. Будь-які дії, що загрожують втратою інформації, повинні бути підтверджені користувачем. Введена користувачем інформація повинна перевірятися програмно, користувач повинен отримувати вичерпну інформацію про свої помилки.

*Забезпечення цілісності даних*. При будь-яких діях користувача не повинні губитися дані або їх цілісність (некоректність індексів, втрата посилань в зв'язках після видалення-додавання записів і так далі).

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

*Терміни та інтерфейс*. У діалогових вікнах повинні використовуватися тільки терміни, зрозумілі користувачеві, і не повинні використовуватися терміни розробника («запис», «індексація» і т. д.). Поява службових англомовних повідомлень неприпустима. У повідомленнях користувачеві слід дотримуватися норм ввічливості, кольорова гамма повинна слідувати загальноприйнятим рекомендаціям.

*Використання клавіатури*. На будь-якому етапі натискання будь-якої клавіші повинно ігноруватися або викликати передбачені дії, описані в інструкції користувача. Прив'язка дій до клавіш повинна бути загальноприйнятою: F1 – допомога; Entег – згода, завершення введення; Еsс – відмова, повернення до попереднього вузла алгоритму (з відновленням екранної форми); Таb – перехід до наступного поля, вікна; Shift-Tab – повернення до попереднього поля.

*Культура кодування.* Код програми повинен задовольняти загально прийнятим вимогам. Вимоги до коду програми наведені в додатку Б.

# ПОРЯДОК ВИКОНАННЯ КУРСОВОЇ РОБОТИ

Основні етапи виконання курсової роботи такі:

1. формулювання мети роботи;
2. опис вимог до програми;
3. проектування;
4. кодування;
5. тестування;
6. завершення пояснювальної записки;
7. захист роботи.

Важливою складовою курсової роботи є пояснювальна записка (далі просто «записка» ). Розділи записки пишуться паралельно з роботою над програмою, або навіть раніше, і послуговують доказом того, що той чи інший етап роботи завершено вчасно.

В записці не повинно бути нічого, що безпосередньо не стосується курсової роботи, тобто, загальних місць, переписаних розділів підручника і т.п.

## Формулювання мети роботи

На цьому етапі виконавець знайомиться з предметною галуззю своєї роботи і пише розділ «Вступ».

У вступі формулюють мету роботи і описують спосіб її застосування. Тут треба роз’яснити, у чому полягає типова діяльність користувача майбутньої програми, і як зміниться ця діяльність після того, як програмою почнуть користуватися.

## Опис вимог до програми

На цьому етапі виконавець визначає перелік функцій, які повинна виконувати програма, і детально їх описує. Перелік не обмежується лише діями у відповідь на запроси користувача, як-от авторизація, пошук, заповнення екранних форм, а включає і потреби системи, як-от зберігання даних на диску, завантаження з диску, перевірка цілісності даних.

Опис кожної функції здійснюється окремо и обов’язково включає ескізне зображення екранних форм та вікон, якщо програма має графічний інтерфейс користувача.

Опис окремої функції повинен бути деталізованим настільки, щоб будь-який програміст, спираючись лише на нього, зміг реалізувати цю функцію програмно, не ставлячи додаткових питань. Опис функцій послуговує також для тестування програми і для оцінювання відповідності програми заявленій меті, тому цей етап роботи є ключовим.

На цьому етапі виконавець визначає сценарії використання програми та описує функції програми. Етап завершується написанням розділу записки «Опис вимог».

Кожен сценарій повинен описувати основну послідовність дій, з яких складається і можливі альтернативні послідовності. Сценарій повинен мати ідентифікатор, назву і передумови виконання. Приклади сценаріїв надані нижче.

**Сценарій 1. Пошук книги**

*Передумова*

Користувач пройшов аутентифікацію і відкрив головне вікно програми.

*Основний сценарій*

1. Користувач заповнює поля на панелі пошуку і натискає кнопку "Search".
2. Програма знаходить книги, які задовольняють усім умовам пошуку.
3. Користувач бачить список знайдених книг.

*Додатковий сценарій*

1. Користувач заповнює поля на панелі пошуку і натискає кнопку "Search".
2. Програма не знаходить жодної книги, яка задовольняє умовам пошуку.
3. Користувач бачить повідомлення, що книг не знайдено.

**Сценарій 2. Додавання книги**

*Передумова*

Бібліотекар пройшов аутентифікацію і обрав в меню пункт додавання книги.

*Основний сценарій*

1. Програма відкриває вікно додавання книги.
2. Бібліотекар заповнює поля у вікні додавання і натискає кнопку "OK".
3. Програма перевіряє коректність даних в полях, створює нову книгу і додає її до списку книг.
4. Програма сповіщає користувача, про успіх операції додавання.

*Додатковий сценарій*

1. Програма відкриває вікно додавання книги.
2. Бібліотекар заповнює поля у вікні додавання і натискає кнопку "OK".
3. Програма встановлює, що дані не коректні.
4. Програма сповіщає користувача, про помилку в даних і про те, яким чином її виправити.

Із усіх сценаріїв використання виводять функції програми.

Опис кожної функції роблять окремо і супроводжують ескізами користувацького інтерфейсу.

Опис окремої функції повинен бути деталізованим настільки, щоб будь-який програміст, спираючись лише на нього, зміг реалізувати цю функцію програмно, не ставлячи додаткових питань. Опис функцій послуговує також для тестування програми і для оцінювання відповідності програми заявленій меті, тому цей етап роботи є ключовим.

Приклад опису однієї функції наведений нижче.

**Функція 1. Додавання книги**

Панель додавання книги розташовується на головному вікні програми (рис 3.1) і містить такі текстові поля.

Назва книги – фрагмент назви книги.

Автори - фрагмент прізвища автора або одного з авторів.

Рік відання – число від 1000 до поточного року.

Ключові слова – довільні рядки, записані через кому.

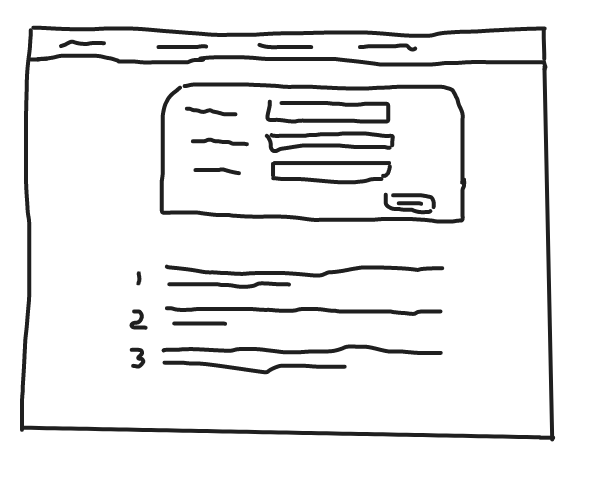


Рисунок 3.1 – Головне вікно програми

В текстових полях регістр не важливий, довжина рядка не перевищує 100 символів. Відсутність даних в полі вводу означає, що відповідна умова у пошуку не застосовується.

В правому нижньому куті панелі міститься кнопка Search.

Повідомлення про помилку вводу буде з'являтися справа від поля з помилкою.

Результат пошуку буде показаний у вигляді списку описів знайдених книг на головному вікні програми нижче панелі пошуку. Опис книги складається з автору, назви і ріку видання книги.

Якщо жодної книги не знайдено, замість списку книг користувач бачить повідомлення: "No book found".

## Проектування

На етапі проектування обирається архітектура (консольна, настільна з GUI, мережева), визначається загальна структура програми, а також формат зберігання зовнішніх даних. Взаємодію частин програми та співвідношення між класами відображують за допомогою діаграм.

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

Завершується етап проектування створенням розділу записки «Проектування програми».

## Кодування програми

На етапі кодування створюється вихідний код програми. Він повинен задовольняти вимогам до вихідного коду, наведеним у додатку Б цих вказівок.

Увесь написаний вихідний код наводять у додатку до пояснювальної записки.

## Тестування програми

Існує два основних різновиду тестування: функціональне та модульне.

Модульне тестування робиться під час кодування и має на меті автоматичну перевірку окремих частин програми. Виконання модульного тестування вельми бажано, але не є обов’язковим.

Функціональне тестування перевіряє, як програма виконує свої функції в умовах коректних та некоректних дій користувача. Воно планується на етапі проектування, виконується у міру того, як реалізуються окремі функції програми і завершується перевіркою програми в цілому.

## Завершення пояснювальної записки

Завершення записки полягає в написанні інструкцій по розгортанню та використанню програми, формулюванні висновків та оформленні кінцевого документа.

В інструкції по розгортанню перераховується програмне забезпечення, яке має бути на цільовому комп’ютері перед встановленням програми, надається покроковий опис дій по встановленню програми, а також опис дій по повному видаленню програми з комп’ютера користувача.

Інструкція користувача пояснює, що треба робити, щоб скористатися кожною з функцій програми. Тут доцільно наводити зображення екранних форм і вікон закінченої програми і надавати покрокові алгоритми дій.

У висновках констатують відповідність програми заявленим цілям, оцінюють ступінь її завершеності, можливі шляхи вдосконалення і розвитку. Можна перерахувати навички, які набув виконавець в процесі розробки.

Вимоги до оформлення пояснювальної записки наведені у наступному розділі.

# ВИМОГИ ДО ОФОРМЛЕННЯ ЗАПИСКИ



## Склад записки

Пояснювальна записка до курсової роботи містить:

* титульний аркуш (додаток В);
* аркуш завдання та календарний план (додаток Г);;
* реферат (додаток Д);
* зміст;
* вступ – 1 сторінка;
* основні розділи (опис вимог, проектування програми, інструкція користувача) – 15-30 сторінок;
* висновки – 1 сторінка;
* перелік джерел посилання (додаток Е);
* додатки.

На титульному аркуші вказують назву дисципліни, по якій виконувалося проектування, тему завдання, прізвище, ім’я та по батькові автора пояснювальної записки. Форма титульного аркушу наведена у додатку Г.

Реферат складається з відомостей про обсяг пояснювальної записки (кількість рисунків, таблиць, додатків, джерел згідно з переліком посилань), тексту та переліку ключових слів. В рефераті відображують мету проекту, отримані результати та застосовані для цього програмні технології. Зразок оформлення реферату наведено у додатку Д.

Основні розділи повинні поділятися на підрозділи і можуть поділятися на пункти. Кожний пункт містить закінчену інформацію.

Обов’язковими розділами записки є: опис вимог, проектування програми, інструкція користувача.

## Загальні вимоги

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

Пояснювальна записка оформлюється у вигляді документу World.

Документ виконується на стандартному аркуші А4 (210x297 мм). Текст і таблиці друкуються чорним кольором, рисунки і схеми чорно-білі або кольорові.

## Вимоги до тексту

Шрифт тексту Times New Roman, прямого накреслення, розмір шрифту 14 пт. Інтервал абзацу полуторний, після та перед рядком інтервал дорівнює 0 пт. Береги: лівий – 3 см., правий – 1,5 см, верхній і нижній – 2 см.

Заголовки структурних розділів (РЕФЕРАТ, ЗМІСТ , ВСТУП, ВИСНОВКИ, ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ) і заголовки основних розділів (1 ОПИС ВИМОГ, 2 ПРОЕКТУВАННЯ ПРОГРАМИ, 3 ІНСТРУКЦІЯ КОРИСТУВАЧА) треба друкувати з абзацного відступу великими літерами напівжирним шрифтом без крапки в кінці. Дозволено їх розміщувати посередині рядка.

Текст розділів складається з речень. В українській мові кожне речення починається з великої літери і закінчується крапкою.

Розділи та підрозділи повинні мати заголовки. Пункти можуть мати заголовки. Заголовки підрозділів і пунктів потрібно друкувати з абзацного відступу з великої літери без крапки в кінці.

Відстань між заголовком і подальшим або попереднім текстом має бути два міжрядкових інтервали (тобто 1 рядок тексту). Приклад оформлення тексту пояснювальної записки наведено у додатку Ж.

Кожний розділ починається з нового аркушу. Не допускається поміщати в кінці аркуша заголовок підрозділу, пункту, якщо після цього на сторінці немає хоча б двох рядків тексту.

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

Просто у тексті напівжирний шрифт та курсив використовувати не бажано. Курсивом можуть оформлюватися елементи формули в тексті для однозначного трактування.

## Нумерація

Нумерація розділів, підрозділів, пунктів робиться арабськими цифрами.

Після номеру розділу, підрозділу, пункту крапку не ставлять.

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

*Приклад*:

1.2 Назва другого підрозділу 1-го розділу

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

*Приклад*:

1.2.1 Назва першого пункту другого підрозділу 1-го розділу

Сторінки звіту нумерують арабськими цифрами, охоплюючи додатки. Номер сторінки проставляють праворуч у верхньому куті сторінки без крапки в кінці. Титульний аркуш входить до загальної нумерації сторінок звіту. Номер сторінки на титульному аркуші не ставляють.

## Переліки

В тексті записки можуть бути використані переліки. Перед переліченням ставлять двокрапку.

Якщо подають переліки одного рівня підпорядкованості, на які у записці немає посилань, то перед кожним із переліків ставлять знак «тире». Наприклад:

* текст першого пункту;
* текст другого пункту;
* текст третього пункту.

Якщо у записці є посилання на переліки, підпорядкованість позначають малими літерами української абетки. Після цифри або літери певної позиції переліку ставлять круглу дужку.

## Оформлення рисунків

Будь які графічні матеріали вважаються рисунками. Рисунки слід розташовувати безпосередньо після тексту, в якому вони згадуються або на наступній сторінці.

На всі рисунки мають бути посилання в записці. Рисунки повинні мати назву, яку розміщують під рисунком по центру сторінки і починають з великої літери.

Рисунки нумерують арабськими цифрами у межах кожного розділу. Їх номер складається з номера розділу та порядкового номера в цьому розділі, якій відокремлюють крапкою.

*Приклад:*

Рисунок 3.1 – Схема алгоритму

В додатках у номер рисунка входить літера додатка.

*Приклад*:

Рисунок А.1 – Модель даних

Посилання на рисунок у тексті записки роблять так: «...на рисунку 3.1 наведена схема алгоритму» або в реченні ставиться посилання на рисунок у круглих дужках, наприклад, «…(див. рис. 3.1 ) ».

Особливу увагу необхідно приділити чіткості ілюстрації, щоб читач міг все побачити, не напружуючи очі

## Оформлення програмного коду

Вихідний код проекту має бути наведений у додатку до пояснювальної записки. Невеликі фрагменти коду за необхідністю можуть бути розміщені в основному тексті записки.

*Приклад*

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;

}

}

## Складання переліку посилань

Перелік джерел посилання складається згідно ДСТУ 8302:2015 [5, 6].

Бібліографічні описи в переліку наводять у порядку першого згадування в тексті. Посилання у тексті на друковані джерела мають подаватися в квадратних дужках, наприклад: [5].

Приклад складання переліку посилань наведено у додатку Е.

## Додатки

Додатки необхідно оформляти як продовження 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 великими літерами повинно бути надруковано слово “ДОДАТОК” і велика буква, що позначає додаток. Додатки слід позначати послідовно прописними буквами українського алфавіту, за винятком букв Ґ, Є, З, І, Ї, Й, О, Ч, Ь. Один додаток позначається як А.

Обов’язковим додатком є вихідний код програми.

# ЗАХИСТ РОБОТИ

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

* наявність діючої програми, яка відповідає проектному завданню;
* правильно оформлений код програми;
* відповідність пояснювальної записки вимогам, викладеним в попередніх розділах;
* виконання вимог, викладених у [«Інструкції для студентів щодо електронного курсового» (створено потрібні каталоги на кафедральному диску, завантажено всі потрібні правильно названі файли, пройдено перевірку на плагіат) .](https://dl.nure.ua/mod/resource/view.php?id=176515)

Захист відбувається перед комісією з трьох викладачів у присутності студентів з академічної групи автора. Списки комісій та керівників за групами наведені у табл.1.

Таблиця 1 – Керівники курсової роботи та члени комісій

|  |  |  |
| --- | --- | --- |
| Група | Керівник | Члени комісії |
| ПЗПІ-21-1, ПЗПІ-21и-1 | Ст. викл. Ляпота В.М. | Ст. викл. Черепанова Ю.Ю., Проф. Бондарєв В.М.,  Ст. викл. Ляпота В.М. |
| ПЗПІ-21-2 | Доц. Кравець Н.С. | Ст. викл. Черепанова Ю.Ю., Доц. Побіженко І. О., Доц. Кравець Н.С. |
| ПЗПІ-21-3 | Ст. викл. Черепанова Ю.Ю. |
| ПЗПІ-21-4 | Ст. викл. Черепанова Ю.Ю. |
| ПЗПІ-21-5 | Ст. викл. Черепанова Ю.Ю. |
| ПЗПІ-21-7 | Доц. Кравець Н.С., Доц. Побіженко І. О. |
| ПЗПІ-21-6 | Проф. Бондарєв В.М. | Проф. Бондарєв В.М., Доц. Побіженко І. О.,  Ст. викл. Черепанова Ю.Ю. |
| ПЗПІ-21-8 | Ст. викл. Черепанова Ю.Ю. |
| ПЗПІ-21-9 | Доц. Побіженко І. О. | Ст. викл. Черепанова Ю.Ю., Доц. Побіженко І. О.,  Ст. викл. Ляпота В.М. |
| ПЗПІ-21-10 | Доц. Побіженко І. О. |
| ПЗПІ-21-11 | Ст. викл. Черепанова Ю.Ю. |

Студент під час доповіді повинен викласти основні результати курсової роботи. Орієнтовний час виступу - 10 хвилин. Після доповіді студент має відповісти на поставлені запитання. Якщо з відповідей на запитання або під час перевірки на плагіат з’ясується, що робота виконана не самостійно, студент отримує незадовільну оцінку.

Курсова робота оцінюється за наступними критеріями:

* якість програмного продукту та інтерфейсу користувача;
* якість вихідного коду;
* виступ та відповіді на запитання;
* якість пояснювальної записки;
* ступінь відповідності термінів виконання етапам календарному плану.

# ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Троелсен, Э, Джепикс, Ф. Язык программирования C# 7 и платформы

.NET и .NET Core, 8-е изд. : Пер. с англ. СПб. : ООО “Диалектика”, 2018

— 1328 с. 978-5-6040723-1-8

1. Бондарев, В.М. Объектно-ориентированное программирование на С#: учеб. пособ. Х.: Компания СМИТ, 2009. 224 с. - ISBN 978-966-2028-34-8.
2. Юркин А.Г. Задачник по программированию. СПб.: Питер, 2002. 192 с.
3. ДСТУ 3008:2015 Інформація та документація. Звіти у сфері науки і техніки. Структура та правила оформлювання. К.: ДП «УкрНДНЦ», 2016. 26 с.
4. ДСТУ 8302:2015. Інформація та документація. Бібліографічне посилання. Загальні положення та правила складання. К.: ДП «УкрНДНЦ», 2016. – 16 с.

# ДОДАТОК A

## Приклади тем курсових робот

Приклади тем взято з книги [3] та допрацьовано У всіх темах передбачити введення, зберігання, зміну інформації.

**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 так далі. 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 праці або навчання, посада знайомих, колег і родичів, характер знайомства, ділові якості і так далі. 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 (при 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 послуги та іншими ознаками. Вивід в 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 в 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 інше. 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 всіх 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 відгрузки (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 (при 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 постояльців, від'їжджаючих сьогодні, звільнення місця або оформлення затримки з випискою додаткової квитанції. Можливість дострокового від'їзду з перерахунком. Пошук гостя по 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 найближчого справи: за поточною датою і часом; видалення вчорашніх справ або перенесення на майбутнє. Аналіз «накладок» - перетинань планованих справ. Перегляд справ на завтра, післязавтра і так далі.

**Термінологія**. База визначень будь-якої науки: вводиться термін, його тлумачення (визначення), посилання на терміни, що використовуються, (по типу 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 та інше; 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 інше.

**Довідник поштової індексації.** Країна, область (край), район, населений пункт, поштовий індекс. Пошук по будь-якої сукупності 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, оціночна 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 і інше), 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 в відеотеці. Підбір фільму для перегляду по ряду критеріїв, підбір. Складання вибірки ряду фільмів за часом відтворення, розміру зовнішнього носія, жанру і акторської і режисерському 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 і антропологічні дані, громадянство, походження, вид спорту, клуб або команда, дані про особисте рекорд або перемогах та інше. Вибір за довільною ознакою. Пошук рекордсмена в заданому виді спорту.

**Довідник радіолюбителя.** Бази паспортних даних транзисторів, діодів, тиристорів та так далі: марка, характеристики, гранично допустимі умови експлуатації, ціна, облік взаємозамінності та інше. Підбір по заданим вимогам.

**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, адресу батьків, громадянська професія, освіта, звання та дата його отримання, посада, підрозділ, форма служби (термінова, кадрова, контрактна та інше), період служби, особливості характеру та ставлення до служби. Формування списків: заданого підрозділу, офіцерського складу, новобранців та інше.

**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 площадь, позачергове та першочергове поліпшення, побажання на район та інші. 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, в 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 джерелам, пошук по темам та ключовим словам. 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 інше). 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 в колекції. Власна колекція.

**Довідник філателіста.** Марки: країна, номінальна вартість, рік випуску, тираж, особливості. Філателісти: країна, ім'я, контактні координати, наявність рідкісних марок в 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 та 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 інше. 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 та сузір'їв в 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 інше.

**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 інше), розділ бібліотеки (спеціальна література, хобі, домашнє господарство, белетристика та інше), походження та наявність книги в даний момент, суб'єктивна оцінка книги. Вибір книг за довільним запитом; інвентаризація бібліотеки.

# ДОДАТОК Б

## Узгодження що до кодування програм

Курсову роботу слід виконувати на мові С#. Нижче викладені вимоги до вихідного коду програми.

1 Структурні узгодження

### 1.1 Структура програмного проекту

Кожен проект повинен розташовуватися в окремому підкаталозі каталогу рішення. Не кладіть файли проекту та файли рішення в один і той же каталог.

Використовуйте підкаталоги для організації вихідного коду проекту. Простори імен повинні відповідати структурі каталогів.

### 1.2 Структура файлу вихідного коду

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

1) using-декларації;

2) декларації простору імен;

3) при необхідності типи *enum* та допоміжні структури, потрібні для

даного класу.

4) оголошення класу. Рекомендується оголошувати один клас в одному

файлі.

### 1.3 Структура класів

Класи та структури повинні оголошуватися в наступному порядку:

1) поля;

2) конструктори;

3) властивості (спочатку public, потім protected, потім private);

4) методи (спочатку public, потім protected, потім private);

5) делегати та події.

1. Форматування коду

Код повинен бути відформатований відповідно до наступних правил.

Параметр "Tabs" повинен бути встановлений в положення "Keep spaces".

Розмір відступу повинен дорівнювати 3 або 4 пробілам.

2.1 Вибір та повторення

Програмні конструкції вибору та повторення повинні бути відформатовані, як в наступних 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;

}

* 1. Порожні рядки

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

* після останнього оголошення "using" перед оголошенням простору імен;
* між методами;
* між оголошенням локальних змінних методу та першою інструкцією методу;
* перед ізольованою в логічному сенсі частиною коду методу.
  1. Пробіли

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

* перед круглими дужками, наступними за ключовим словом

if (condition)

* після коми в списку аргументів

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

* між бінарним оператором та його операндами:

int result = **argumentOne + argumentTwo**;

* між частинами інструкції "for":

for (int i = 0; i < MAX\_ELEMENTS; ++i)

Аргументи унарних операторів пробілами не відокремлюються.

i++;

--counter;

2.4 Переноси рядків

Довгі рядки повинні бути згорнуті. Довжина рядка не повинна перевищувати 80 символів. При згортанні довгих рядків дотримуйтесь наступних правил:

* переносіть рядок після коми;
* переносіть рядок перед оператором;
* віддавайте перевагу «логічному» розбиттю рядка, не «фізичному»;
* 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;

}

Довгий список параметрів методу також повинен бути згорнутий, так само як та список аргументів у інструкції виклику методу:

Не оголошуйте декілька змінних в одному рядку, використовуйте окремий рядок для кожної змінної.

### Угоди про найменування

Не використовуйте угорську або іншу префіксну нотацію.

Імена локальних змінних та аргументів методів повинні записуватися в нотації «верблюд»:

string customerName;

private int DoSomething(

int **firstArgument,**

float **secondArgument**);

Імена закритих полів класу повинні слідувати нотації «верблюд».

private string **customerName**;

Імена відкритих полів класу повинні слідувати нотації «Паскаль».

public int **CustomerID;**

Імена властивостей та методів класу повинні слідувати нотації «Паскаль».

|  |
| --- |
| public bool **ValidateAmount**();  public bool **FileExists** {  get;  } |

Імена класів, структур та нумераторів повинні слідувати нотації «Паскаль».

public class **CustomerAccount**

{

}

private enum **WorkMode**

{

**Add,**

**Update**;

}

Імена методів створюють за шаблоном «дієслово + іменник», наприклад, *UpdateAccount*.

Використовуйте однину, а не множину в іменах нумераторів *enum*. Інакше кажучи, *WorkMode* – це правильне ім’я, а *WorkModes* – ні.

Використовуйте значущі імена навіть для закритих методів, властивостей, типів та інше. Уникайте коротких імен, таких як *a, b, n* за винятком загальноприйнятих імен для змінних циклу, таких як *i* або *j*.

### Культура кодування

Дублювання коду в програмі забороняється.

Обов'язкова заміна числових та строкових літералів символічними константами. Виняток можна зробити для самоочевидних констант, таких як 0 та 1. Давайте символічним константам значущі імена.

Рекомендується об’єднувати пов’язані між собою константи цілих типів в нумератори, а не цілих типів – в абстрактні класи з відкритими статичними константними членами.

Уникайте довгих та складних методів, розбивайте їх на декілька коротких. При модифікації коду максимально використовуйте рефакторинг.

Завжди звертайте увагу на попередження компілятора. Подбайте, щоб їх не було.

### Коментарі

Обов'язковими є коментарі до класів, до інших відкритих типів та до всіх відкритих членів класів.

Коментарі всередині коду методів не бажані. Код повинен легко сприйматися і без коментарів.

Коментар не повинен перефразовувати те, що написано в коді. Намагайтеся давати змістовні пояснення. Наприклад, коментар *Increment i by one* до рядка 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 повинна бути прописною, в кінці речення ставте крапку.

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

Коментарі повинні підкорятися загальним правилам згортання рядків.

Не використовуйте коментарі в стилі C: /\* … \*/

### Специфіка Windows Forms

Дотримуйтесь *Microsoft User Interface Guidelines*, якщо до вас не пред'являють інших вимог.

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

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

Використовуйте нотацію «верблюд» для найменування екземплярів керуючих елементів на формах, наприклад, *optionsHeading*.

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

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

Джерело інформації:

https://docs.microsoft.com/en-us/dotnet/standard/design-guidelines/

# ДОДАТОК В

## Зразок титульного листа

Міністерство освіти і науки України

Харківський національний університет радіоелектроніки

Кафедра програмної інженерії

КУРСОВА РОБОТА

ПОЯСНЮВАЛЬНА ЗАПИСКА

з дисципліні “ Об’єктно -орієнтоване програмування”

<НАЗВАНИЕ>

Керівник , <посада>  <прізвище, ініціали > Студент гр. <шифр групи> <прізвище, ініціали >

Комісія:

<посада> \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ <прізвище, ініціали >

<посада> \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ <прізвище, ініціали >

<посада> \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ <прізвище, ініціали >

Харків 2022

# ДОДАТОК Г

## Аркуш завдання та календарний план

ХАРКІВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ РАДІОЕЛЕКТРОНІКИ

Кафедра *програмної інженерії*

Рівень вищої освіти *перший (бакалаврський)*

Дисципліна *Об’єктно-орієнтоване програмування*

Спеціальність *121*  *Інженерія програмного забезпечення*

Освітня програма: *Програмна інженерія*

Курс 1 .Група *ПЗПІ-21 -Х* . Семестр *2* .

***ЗАВДАННЯ***

***на курсовий проект студента***

(Прізвище, Ім'я, По батькові)

1. Тема проекту:

1. Термін здачі студентом закінченого проекту: ***“24” - червня - 2022 р***.
2. Вихідні дані до проекту:

*Специфікація програми, методичні вказівки до виконання курсової*

*роботи*

1. Зміст розрахунково-пояснювальної записки:

*Вступ,* *специфікація програми, проектна специфікація, інструкція користувача, висновки\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

*\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

КАЛЕНДАРНИЙ ПЛАН

|  |  |  |
| --- | --- | --- |
| *№* | *Назва етапу* | *Термін виконання* |
| 1 | Видача теми, узгодження і затвердження теми | 16.02.2021 - 14.03.2022 р. |
| 2 | Формулювання вимог до програми | \_\_.\_\_.2022 – \_\_.\_\_.2022 р. |
| 3 | Розробка підсистеми зберігання та пошуку даних. | \_\_.\_\_.2022 – \_\_.\_\_.2022 р. |
| 4 | Розробка функцій ….. | \_\_.\_\_.2022 – \_\_.\_\_.2022 р. |
| 5 | Розробка функцій зберігання та завантаження даних | \_\_.\_\_.2022 – \_\_.\_\_.2022 р. |
| 6 | Тестування і доопрацювання розробленої програмної системи. | \_\_.\_\_.2022 – 14.05.2022 р. |
| 7 | Оформлення пояснювальної записки, додатків, графічного матеріалу | \_\_.\_\_.2022 – 09.06.2022 р. |
| 8 | Захист | 06.06.2022 – 24.06.2022 р. |

Студент \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Керівник \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(Прізвище, Ім'я, По батькові)

« 16 »\_лютого\_\_\_\_\_\_\_\_\_\_ 2022 р.

# 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», яка буде надавати користувачу довідки про товари та магазини.

В результаті отримана програма, що 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.

В процесі розробки використано середовища Microsoft Visual Studio 2022, фреймворк Windows Forms, платформи .NET 6.0, мова програмування C#

ПОКУПЕЦЬ, МАГАЗИН, ЗВІТ, ООП, .NET, МОВА C#

# ДОДАТОК Е

## Приклад оформлення переліку джерел посилання

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Андреев, В. В. Как организовать делопроизводство на. М.: ИНФРА-М, 1997. 94 с.
2. Белов А. В., Николаев В. Н. Финансы и кредит: учеб. - К.: Университет, 2004. 215 с.
3. Агафонова Н. Н., Богачева Т. В. , Глушкова Л. И. Гражданское право: учеб. пособие / под общ. ред. А. Г. Калпина ; МОН Украины. - 2-е изд., перераб. и доп. - Х. : Фактор, 2000. 542 с.
4. Височанський В. С. , Кардаш А. І., Костєв В. С. та ін. Елементи інформатики: довідник. К. :Наук. думка, 2003. 92 с.
5. Документация по .NET. URL: https://docs.microsoft.com/ruru/dotnet/#pivot=docs&panel=getstarted (дата звернення: 26.02.2020).
6. Microsoft Developer Network. Библиотека MSDN / MSDN – сеть разработчиков Microsoft. URL: https://msdn.microsoft.com/ru-ru/ (дата звернення: 26.02.2020).