

**Київський національний торговельно-економічний
університет
Кафедра цифрової економіки та системного аналізу**

АЛГОРИТМІЗАЦІЯ ТА ПРОГРАМУВАННЯ

**МЕТОДИЧНІ РЕКОМЕНДАЦІЇ
до виконання курсових робіт**

освітній ступінь

«бакалавр»

**галузь знань
спеціальність
спеціалізація**

**12 «Інформаційні технології»
124 «Системний аналіз»
«Системний аналіз»
«Інформаційні технології та бізнес-
аналітика (Data Science)»**

Київ – 2019

Розповсюдження і тиражування без офіційного дозволу КНТЕУ заборонено

Автор: В. В. Кулаженко, канд. екон. наук, ст. викл.

Розглянуто і схвалено на засіданні кафедри економічної кібернетики «11» березня 2019 р., протокол № 8.

Рецензенти: викладач
практик

МЕТОДИЧНІ РЕКОМЕНДАЦІЇ ДО ВИКОНАННЯ КУРСОВИХ РОБІТ

освітній ступінь

«бакалавр»

галузь знань
спеціальність
спеціалізація

12 «Інформаційні технології»
124 «Системний аналіз»
«Системний аналіз»
«Інформаційні технології та бізнес-аналітика (Data Science)»

Автор: КУЛАЖЕНКО Володимир Валерійович, канд. екон. наук, ст. викл.

ЗАГАЛЬНІ ПОЛОЖЕННЯ

Відповідно до навчального плану, нормативних документів КНТЕУ, у тому числі «Положення про організацію виконання та захисту курсових робіт (проектів) у КНТЕУ» студенти спеціальностей «Системний аналіз» та «Інформаційні технології та бізнес-аналітика (Data Science)» освітнього ступеня бакалавр виконують курсову роботу з дисципліни «Алгоритмізація та програмування».

Метою підготовки курсової роботи є поглиблення теоретичних і практичних знань студентів у предметній галузі «Алгоритмізація та програмування» та їх систематизація; набуття досвіду самостійного аналізу наукових праць та програмних додатків вітчизняних та зарубіжних фахівців; вивчення і аналіз питань, пов'язаних з різними аспектами проектування та реалізації програмних додатків; методики та інструментарію написання програмних продуктів, а також їх тестування; формування культури пошуку актуальних досягнень та надбань у цій сфері та вміння самостійно застосовувати їх для вирішення прикладних проблем.

Виконання курсової роботи з дисципліни «Алгоритмізація та програмування» та її захист є формою контролю рівня знань студентів з цієї навчальної дисципліни.

Під час роботи над курсовою студент має знати:

- основні поняття та терміни, що використовуються у сфері програмування;
 - особливості проектування програмного продукту та окремих його функціональних модулів;
 - функціональні можливості основних модулів (бібліотек) обраної мови програмування;
 - основні алгоритми та шаблони об'єктно-орієнтованого програмування.
- вміти:
- аналізувати, узагальнювати, досліджувати літературні джерела та інформаційні бази даних;
 - працювати з науковими статтями, монографіями, а також з методичними, інструктивними матеріалами, критично аналізувати та виявляти їхні позитивні й негативні аспекти;
 - використовувати сучасний інструментарій реалізації програмних продуктів, які складаються з оптимізованих алгоритмічних структур об'єктно-орієнтованої парадигми програмування;
 - писати програмний код, який відповідає сучасним стандартам

іменування об'єктів, коментування та логічного структурування;

- проводити якісне тестування власних програмних розробок;
- використовувати математичні, статистичні, експертні та інші способи оброблення інформації;
- узагальнювати й аналізувати фактичний матеріал, робити відповідні як теоретичні, так і прикладні висновки;
- логічно й чітко викладати матеріал, давати ґрунтовні оцінки різним явищам і процесам;
- Швидко знаходити та опановувати інструментарій, призначений для вирішення поточної задачі;
- оформляти результати наукового дослідження відповідно до існуючих стандартів.

Курсова робота є результатом самостійної роботи студента за обраною темою. Матеріал роботи має бути логічно-послідовно викладений, зі стислістю і точністю формулювань, практичною спрямованістю рекомендацій, чіткими висновками. Програмний додаток, який є результатом виконання курсової роботи, повинен вирішувати поставлене завдання, мати інтуїтивно-зрозумілий інтерфейс, інструкцію для використання, оптимізований, чіткий та зрозумілий програмний код, написаний у об'єктно-орієнтованому стилі.

Наукове керівництво курсової роботи здійснюється викладачами кафедри кібернетики та системного аналізу. Керівник проводить індивідуальні консультації для студентів за відповідним графіком, встановленим на кафедрі.

Етапи виконання курсової роботи:

- Вибір теми.
- Вивчення літературних джерел з обраної теми.
- Збирання та оброблення інформації.
- Виконання практичної частини.
- Обґрунтування висновків та пропозицій.
- Написання та оформлення роботи.
- Подання на кафедру та захист.

Виконання роботи здійснюється в межах одного семестру. Кожний студент самостійно обирає тему дослідження курсової роботи (із запропонованих у тематиці). Він може також запропонувати власну тему не передбачену у переліку, але близьку за спрямованістю.

Студент має виконати курсову роботу згідно з графіком та вчасно подати її на кафедру. Курсові роботи, подані на кафедру з порушенням встановлених графіком строків без поважних причин не рецензуються і повертаються студентам. Якщо курсова не виконана в строк з поважних причин, то за заявою студента деканат встановлює індивідуальний

термін її рецензування, обов'язковий як для студента, так і для керівника.

ВИБІР І ЗАТВЕРДЖЕННЯ ТЕМИ. СКЛАДАННЯ ПЛАНУ КУРСОВОЇ РОБОТИ

Робота над курсовою починається з вибору теми, яка має бути актуальною, відповідати профілю дисципліни «Алгоритмізація та програмування» бакалаврської програми. Тематика курсових робіт (орієнтовна) пропонується кафедрою, а список тем щорічно оновлюється.

Виконання курсової роботи з однієї теми кількома студентами однієї групи не припустиме. Для затвердження теми курсова робота студент подає письмову заяву на ім'я завідувача кафедри (додаток В). Заява про затвердження теми та картка контролю обов'язково додається до виконаної курсової роботи.

Обрані студентами й узгоджені з науковими керівниками теми робіт затверджуються на засіданні кафедри. Курсові роботи виконані за темами, що не затверджені кафедрою, не розглядаються.

Після вибору теми студент розробляє завдання з курсового проекту і узгоджує його з науковим керівником. Після цього студент за допомогою наукового керівника може розпочати роботу над курсовою роботою. Насамперед конкретизується призначення дослідження, тобто чітко формулюється його завдання. До плану курсової можна вносити зміни, але їх обов'язково слід погодити з керівником.

Для ефективної роботи студенту рекомендується угоджувати свою курсову роботу з науковим керівником поетапно, тобто при написанні кожного наступного підрозділу.

Перед передачею керівнику будь-якого підрозділу у чистовому чи чорновому варіанті, студент зобов'язаний перевірити текст на дотримання вимог оформлення, а також на граматичні помилки за допомогою редактора MS Word.

ВИМОГИ ДО ЗМІСТУ ТА ОБСЯГУ КУРСОВОЇ РОБОТИ

Курсова робота, як оригінальне теоретично-прикладне дослідження, має бути логічно побудоване, з послідовним викладом і завершеністю. Для успішного виконання роботи слід чітко дотримуватись основних вимог до її теоретичного рівня, змісту, структури, обсягу, форми викладення матеріалу, оформлення і захисту. Загальний обсяг курсової роботи має становити близько 28-33 сторінок друкованого тексту (не враховуючи титульний аркуш, список використаних джерел і додатки).

Робота має бути виконана з урахуванням державних і галузевих стандартів (ДСТУ 3008-95. Документація. Звіти у сфері науки і техніки. Структура та правила оформлення).

Мова курсової роботи – державна, стиль – науковий, чіткий, без орфографічних і синтаксичних помилок. Наприкінці кожного розділу слід наводити висновки, що підсумовують проведене дослідження.

У разі використання студентом аббревіатур та скорочень, слід перед змістом слід додати розділ «Перелік аббревіатур та умовних скорочень».

Загальна структура курсової роботи є такою:

- *титульний аркуш*, який оформлюється за наведеним зразком (див. дод. А);
- *анотація* (0,5–1 сторінка), в якій слід зазначити загальний обсяг роботи, кількість таблиць, рисунків, ключові поняття і викласти в дуже стислій формі сутність роботи. Анотація має бути поданою українською та іноземною (не російською) мовами;
 - *зміст роботи*;
 - *список використаних аббревіатур та скорочень*;
 - *вступ*;
 - *основний текст*;
 - *висновки*;
 - *список використаних джерел*;
 - *додатки*.

Зміст курсової роботи розміщують на 1–1,5 сторінках. В ньому наводяться назви всіх розділів і підрозділів (параграфів) із зазначенням початкових сторінок. План роботи має відображати сутність проблеми, її складність і логіку дослідження. Назви розділів і підрозділів мають бути стислими і зрозумілими, літературно грамотними, тісно пов'язаними з назвою роботи, але не повторювати її. Зміст повинен бути інтерактивним та розробленим за допомогою відповідного пункту меню MS Word.

У **вступі** (2–3 сторінок) дається наукове обґрунтування вибору даної теми. Повинні бути чітко висвітлені актуальність даної тематики, формулюються мета, завдання, об'єкт та предмет дослідження, наводиться перелік застосованих методів дослідження, ступінь розробки проблеми українськими та зарубіжними вченими та фахівцями.

Теоретична частина (до 14 сторінок) складається з наступних елементів:

- постановка задачі та її опис;
- аналіз існуючих вітчизняних та зарубіжних програмних

рішень даної проблеми;

- виявлення особливостей, взаємозв'язків і кількісних закономірностей, аналіз вхідної та вихідної інформації;
- побудова математичної (або інформаційно-логічної) моделі програмного додатку;
- опис алгоритму.

У процесі постановки задачі необхідно виокремити поставлену проблему, дослідити джерела її витоків, поточний стан та тенденції подальшого розвитку. Разом з задачею, студент повинен ставити конкретні завдання перед собою, сумарний результат яких зміг би в певному обсязі вирішити дану проблему.

Аналіз існуючих рішень є важливим етапом, оскільки вивчення існуючих програмних рішень визначає напрямок роботи та є основою для розробки інтерфейсу та функціональних можливостей програмного додатку. В залежності від теми курсової роботи, дана частина може бути виконана у вигляді порівняльної таблиці.

Побудова моделі майбутнього програмного рішення також є обов'язковим. В залежності від характеру задачі, модель може бути математичною або інформаційно-логічною.

Математична модель – це постановка задачі, «перекладена» на математичну мову. Для цього з постановки задачі виділяють параметри, які відомі за умовою і змінні величини, які, як правило, необхідно визначити. Далі, за допомогою математичних співвідношень будуть математичну модель задачі (цільову функцію, систему обмежень на допустимі розв'язки задачі, обмеження на змінні параметри, тощо). Прикладом програмного додатку, для якого математична модель обов'язкова, є реалізація задач нейромережевого моделювання, аналітичні додатки тощо.

Інформаційно-логічна модель відображає дані предметної області у вигляді сукупності об'єктів та зв'язків між ними. Як правило, такі моделі будуються при створенні систем управління базами даних (СУБД), де суб'єктом завжди є користувач, а об'єктом – оптимізація процесів управління (зберігання, обробка, видача користувачу) даними. Прикладом програмного додатку такого типу є 1С MS Access, Парус тощо.

Студент, за бажанням, може обрати таку мову програмування, яка буде на його думку краще вирішувати поставлені завдання.

При описі алгоритму роботи програмного додатку, потрібно зазначити основні зв'язки, за якими буде функціонувати додаток, послідовність виконання його функцій, тощо. На основі інформації, закладеної у даному підрозділі, буде побудована UML-діаграма.

Практична частина (до 14 сторінок) складається з наступних елементів:

- UML-діаграма структури програмного продукту;
- аналіз тексту програмного додатку;
- опис користувацького інтерфейсу;
- текст програми, з коментарями до кожного блоку чи процедури;
- тестовий набір даних та контрольний приклад;
- результати розрахунків за допомогою створеної програми.

UML — уніфікована мова моделювання, яка використовується у парадигмі об'єктно-орієнтованого програмування (рис. 1). Вона є невід'ємною частиною уніфікованого процесу розробки програмного забезпечення, використовує графічні позначення для створення абстрактної моделі системи.

В такій діаграмі повинні знайти відображення всі структурні об'єкти програмного додатку, а також їх важливі методи та властивості, які відповідають за взаємодію класів між собою та безпосереднє виконання поставленої задачі.

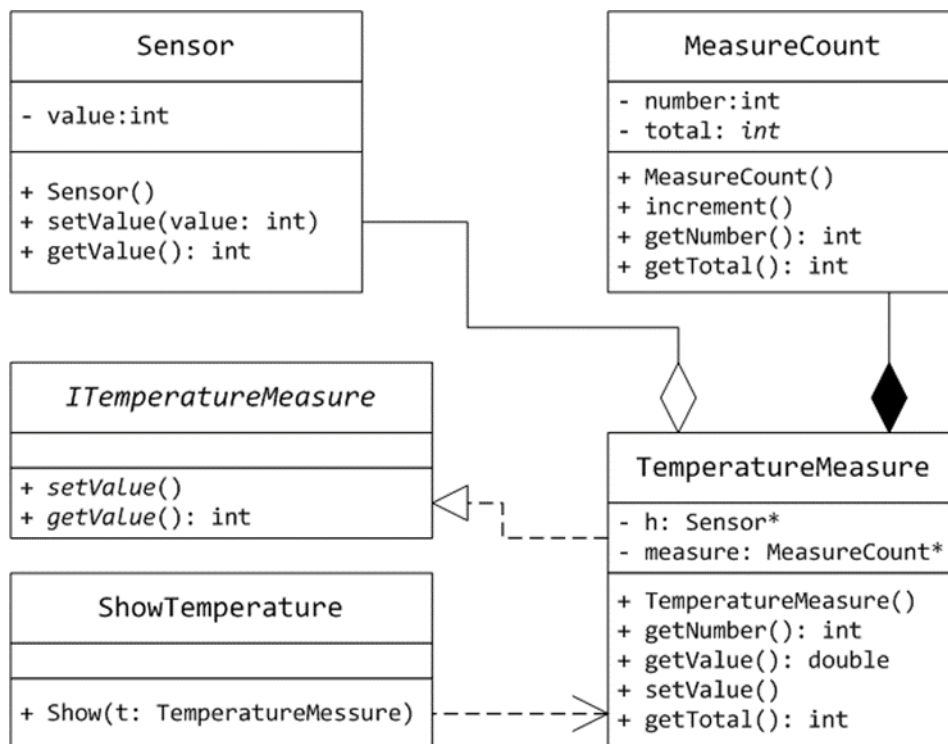


Рис. 1. Приклад UML-діаграми

Аналіз тексту програми описує інструментарій, який реально був використаний для реалізації програмного додатку. Він включає в себе перелік методів та модулів (бібліотек), що не входять у стандартну комплектації обраної мови програмування, а також їх коротку характеристику (дані, які приймаються, функціонал, дані, які

повертаються). Рекомендується описувати даний підрозділ стисло, у вигляді таблиці.

Програмний код повинен бути розміщений у додатках курсової роботи у повному обсязі, у читабельному вигляді. Код повинен мати чітку структуру та стилістику.

У курсовій роботі у повній мірі повинен бути розглянутий весь функціонал створеного користувацького інтерфейсу. Він може складатись зі скріншотів програмного додатку та коментарів до них.

За бажанням, студент може підготувати даний підрозділ у вигляді записаного відео, в якому глядач буде ознайомлений з особливостями інтерфейсу (відеоряд повинен мати голосовий аудіо-супровід). Тривалість даного відео не повинна перевищувати 2 хвилин. Відео-файл потрібно прикласти до курсової роботи, або передати науковому керівнику для розміщення на загальнодоступному Інтернет-ресурсі на весь час зберігання курсової роботи (в цьому випадку, в курсову роботу вставляється посилання на відео-файл). Дане відео в обов'язковому порядку повинен демонструватись під час захисту.

Для тестування програми слід навести декілька прикладів, які всебічно характеризують функціональні можливості програми (проекту). У випадку, коли вивід результатів програми займає більше двох сторінок, дозволяється не наводити у друкованому варіанті проміжні результати обчислень для всіх прикладів. У цьому випадку потрібно вибрати найбільш показовий приклад і повністю навести вивід його результатів, а для решти прикладів обмежуватись лише остаточними результатами.

Висновки (2-3 сторінки) є завершальною частиною курсової роботи. Вони містять стислий виклад актуальності теми, результатів використання розробленого додатку, а також пропозиції автора щодо подальшої роботи над ним. Ознайомлення з текстом висновків має сформувані у читача уявлення про ступінь реалізації автором курсової роботи, поставленої мети і завдань.

Ця частина курсової роботи повинна містити певні узагальнення у вигляді коротких тез, основних висновків, практичних розрахунків та пропозицій. Вона не повинна містити нові матеріали, що не були висвітлені в розділах курсової роботи.

Список використаних джерел включає складений за існуючими правилами перелік літературних джерел, які використовуються для підготовки курсової роботи (дод. Б).

У **додатках** містяться весь код створеного додатку, громіздкі (тобто такі, що не вміщаються на 1 аркуш) та допоміжні таблиці, блок-схеми, зразки форм, таблиць та анкет соціологічного опитування тощо.

ПРАВИЛА ОФОРМЛЕННЯ РОБОТИ

Курсова робота має бути виконана й оформлена з додержанням усіх технічних вимог до наукових робіт. Текст має бути набраний на комп'ютері в текстовому редакторі *MS Word XX* для *MS Windows XX* на одному боці аркушів білого паперу формату А4. Шрифт Times New Roman, 14 пт, з 1,5 міжрядковим інтервалом та відступом у новому абзаці від лівого краю у 1,25 см. Інтервали перед та після абзаців мають бути відсутні. Можна також подати таблиці та ілюстрації на аркушах формату А3.

Текст розміщується на сторінці, яка обмежується полями: лівим – не менше 30 мм, правим – не менше 10 мм, верхнім – не менше 20 мм, нижнім – не менше 20 мм. Відстань між заголовком – 15–20 мм.

Список використаних аббревіатур і скорочень обов'язково має бути окремим підрозділом і передувати викладенню основної частини курсової роботи.

Скорочення, символи та терміни розміщуються у два стовпчика: ліворуч – символи, аббревіатури та спеціалізовані терміни, праворуч – їхнє тлумачення.

Назви розділів і параграфів у змісті й тексті мають бути однаковими. Відгук наукового керівника і зовнішню рецензію не нумерують. Вступ, кожний розділ, висновки і список літератури починаються з нової сторінки, а наступний підрозділ – одразу після закінчення попереднього.

Розділи, підрозділи, пункти й підпункти слід нумерувати арабськими цифрами та друкувати з абзацним відступом.

Розділи мають мати порядкову нумерацію у межах усього тексту за винятком додатків (1, 2, 3 і т.д.).

Номер підрозділу або пункту включає номер розділу і порядковий номер підрозділу або пункту, відокремлені крапкою (1.1, 1.2 і т.д.).

Після номеру розділу, підрозділу в тексті роботи крапку не ставлять.

Якщо текст поділяють тільки на пункти, то їх слід нумерувати (за винятком додатків) порядковими номерами в межах усього тексту.

Заголовки

Розділи, підрозділи повинні мати заголовки, що чітко й коротко відображають їх зміст. Заголовки розділів, підрозділів слід друкувати з абзацним відступом з великої літери без крапки в кінці та без

підкреслень.

Якщо заголовок складається з двох речень, то їх відокремлюють крапкою. Переноси в словах у заголовках розділів не допускаються. При використанні набірних друкарських форм заголовки розділів і підрозділів слід виділяти шрифтом.

Переліки

Всі переліки в тексті повинні будуватись за допомогою відповідного меню MS Word. Перелік пунктів або підпунктів може бути позначений арабськими цифрами, дефісом (або іншим маркером; слід уникати використання декількох видів маркерів у курсовій роботі) або малою літерою з дужкою. Для подальшої деталізації переліку необхідно використовувати арабські цифри, після яких ставити дужки.

Перелік першого рівня деталізації друкують малими літерами з абзацного відступу, другого рівня – з відступом відносно місця розташування переліків першого рівня.

Приклад:

- a) _____
 - б) _____
 - 1) _____
 - 2) _____
- _____ –

Таблиці

Таблиці використовують для уточнення та зручності порівнювання показників. Їх слід розташовувати безпосередньо після тексту, в якому вона згадується вперше, або на наступній сторінці. Назва таблиці має точно і стисло відображати її зміст. Назву слід розміщувати один раз зліва над першою частиною таблиці, над іншими частинами таблиці пишуть «Продовження табл.» із зазначенням номера таблиці.

Таблиці (за винятком таблиць у додатках) слід нумерувати арабськими цифрами порядковою нумерацією в межах розділу. Номер таблиці складається з номеру розділу і порядкового номера таблиці, відокремлених крапкою, наприклад, «Таблиця 2.1» – перша таблиця другого розділу.

(назва таблиці)

					Заголовки стовпчиків
					Підзаголовки стовпчиків
1	2	3	4	5	Нумерація стовпців
					Рядки

Таблиці кожного додатка позначають нумерацією арабськими цифрами з додаванням перед цифрою позначення додатка.

Якщо в тексті роботи є тільки одна таблиця, то її позначають «Таблиця 1» або «Таблиця В.1» (якщо таблицю наведено у додатку В).

На всі таблиці мають бути посилання в тексті, які складаються зі слова «таблиця» із зазначенням її номера.

Заголовки стовпців і рядків таблиці слід друкувати з великої літери, підзаголовки стовпців – з малої, якщо вони є продовженням заголовка, або з великої, якщо вони мають самостійне значення. У кінці заголовків і підзаголовків таблиць крапки не ставлять, заголовки і підзаголовки стовпців друкують в однині.

Розділення заголовків і підзаголовків боковика і стовпців діагональними лініями не допускається.

Горизонтальні та вертикальні лінії, що розмежовують рядки таблиці, можна не креслити, якщо їх відсутність не ускладнює користування.

Заголовки стовпців, як правило, друкують паралельно рядкам таблиці. За потреби допускається перпендикулярне розміщення заголовків стовпців.

Допускається розміщення таблиці на сторінці з альбомною орієнтацією.

Якщо рядки або стовпці таблиці виходять за формат сторінки, то її ділять на частини, які розміщують одна під одною або поряд. При цьому в кожній частині таблиці повторюють її шапку й боковик.

У разі поділу таблиці на частини допускається її шапку або боковик замінити відповідно номерами стовпців і рядків. При цьому нумерують арабськими цифрами стовпці та (або) рядки першої

частини таблиці.

Якщо в кінці сторінки таблиця переривається і її продовження буде на наступній сторінці, то в першій частині таблиці нижню горизонтальну лінію, що обмежує таблицю, не креслять. Перед наступною частиною таблиці пишуть "Продовження таблиці", після чого вказують номер таблиці за правилами, наведеними вище.

Реалізація програмного додатку

Інтерфейс програмного додатку повинен відповідати завданням, поставленим перед студентом. Вибір бібліотеки (модуля) для його створення, студент узгоджує з науковим керівником.

Створюючи даний програмний додаток, студент повинен розуміти, що кінцевим користувачами таких додатків є, як правило, спеціалісти інших напрямків. Тому основними вимогами до інтерфейсу є його інтуїтивна зрозумілість, цілісність, зручність. Кількість віджетів не повинно бути надмірним для сприйняття.

Крім того, користувацький інтерфейс повинен бути виконаний на українській або іншій міжнародній мові.

Програмний додаток повинен містити користувацьку інструкцію, інтерактивні та інші підказки, що допоможуть користувачу самостійно навчитись працювати з даним продуктом.

Програмний код повинен мати чітку логічну структурованість. Важливим є написання всієї програми у одній стилістиці (логіка іменування об'єктів, змістовне наповнення методів та функцій тощо). Імена об'єктів повинні відображати своє призначення. Кожний блок коду, процедура(метод, функція), клас і т.д. повинні супроводжуватись короткими та зрозумілими коментарями щодо їх змісту та характеру використання.

Написання коду повинно виконуватись згідно об'єктно-орієнтовної парадигми програмування.

Програмний додаток оцінюється за такими критеріями:

- виконання поставлених завдань;
- легкість освоєння інтерфейсу для користувачів;
- якість проектування додатку;
- легкість «читання» програмного коду.

Код програмного додатку у повному обсязі розміщується в додатках курсової роботи.

Графічний матеріал

Графічний матеріал – рисунки (схеми, діаграми, тощо) розміщують у курсовій роботі для встановлення властивостей або характеристик об'єкта, а також для кращого розуміння тексту, в якому мають бути посилання на нього. Графічний матеріал розміщують безпосередньо після тексту, в якому про нього згадується вперше або на наступній сторінці, а за потреби – у додатку.

Таблиці, що доповнюють графічний матеріал, наводять після нього.

Графічний матеріал може мати тематичну назву, яку розміщують під ним.

За потреби під графічним матеріалом наводять пояснювальні дані. Слово «рисунок» і назву подають після пояснювальних даних.

Графічний матеріал (за винятком графічного матеріалу додатків) слід нумерувати арабськими цифрами порядковою нумерацією в межах розділу. Якщо рисунок перший у другому розділі, його позначають «Рис. 2.2».

Код країни	Найменування країни
1001	Туреччина
1002	Іспанія
1003	Туніс
*	0

Рис. 2.2. Діаграма продажу товарів за регіонами

Номер рисунка складається з номерів розділу та порядкового номера рисунка, відокремлених крапкою (рис. 2.2).

Графічний матеріал кожного додатка нумерують арабськими цифрами, перед якими наводять позначення додатка (рис. В.3).

Рисунок (діаграму, схему, тощо), як правило, наводять на одній сторінці. Якщо рисунок не вміщується, то можна перенести його на інші сторінки.

Формули

Формули, за винятком тих, які є в додатках, нумеруються арабськими цифрами у межах розділу, які друкують на рівні формули праворуч у круглих дужках.

Номер формули складається з номера розділу і порядкового номера формули, відокремлених крапкою.

Формули друкуються виключно за допомогою MS Microsoft

Equation або математичних формул.

Посилання в тексті на порядкові номери формули наводять у дужках.

Приклад:

$$\frac{d_y}{d_x} * \frac{x}{y} = a_1, \quad (1.1)$$

Формули в додатках нумерують арабськими цифрами в межах кожного додатка з наведенням перед цифрою позначення додатка.

У формулі як символи фізичних величин слід застосовувати позначення, встановлені відповідними стандартами або іншими документами.

Пояснення символів і числових коефіцієнтів, що входять до формули, якщо вони не пояснювалися в тексті, мають бути наведені безпосередньо під формулою. Пояснення давати з нового рядка в тій послідовності, в якій символи наведено у формулі. Перший рядок пояснення має починатися словом «де» без двокрапки.

Формули, що подаються одна за одною і не розділені текстом, відокремлюють комою.

Правила цитування та використання посилань на використані джерела

При написанні курсової роботи студент має надавати посилання на джерела, матеріали, ідеї, програмний код або висновки, які наводяться в курсовій роботі. Такі посилання дають змогу відшукати документи і перевірити достовірність відомостей про цитування документа, надають необхідну інформацію щодо нього, допомагають з'ясувати його зміст, мову тексту, обсяг. Посилатися слід на останні видання публікацій (тобто такі, що видані не більше 10 років тому). На більш ранні видання можна посилатися лише тоді, коли в них є матеріал, який не включено до сучасних видань.

Якщо використовують відомості, матеріали з монографій, оглядових статей, інших джерел з великою кількістю сторінок, тоді в посиланні необхідно точно вказати номери сторінок, ілюстрацій, таблиць, формул з джерела, на яке дано посилання в курсовій роботі.

Посилання додаються одразу після закінчення цитати у квадратних дужках, де вказується порядковий номер джерела у списку літератури та відповідна сторінка джерела (наприклад [4, с. 35]), або під текстом цієї сторінки у вигляді виноски, в якій вказують прізвище

та ініціали автора, назву джерела, видавництво, рік видання та сторінку, у випадку, якщо це інтернет-ресурс – назву веб-сторінки та її електронну адресу.

Додатки

Матеріал, що доповнює положення курсової роботи, може бути розміщений в додатках у вигляді: графічних матеріалів, таблиць великого формату, розрахунків, програмного коду, описів алгоритмів і програм задач тощо.

Додатки можуть бути обов'язковими (програмний код розробленого студентом додатку) та інформаційними. Інформаційні додатки мають рекомендований або довідковий характер.

Додатки позначають великими літерами української абетки, починаючи з А (за винятком літер Г, Є, З, І, Ї, Й, О, Ч, Ђ). Після слова «Додаток» друкують літеру, що позначає його послідовність.

Допускається позначення додатків літерами латинської абетки за винятком літер І та О.

У разі повного використання літер української та латинської абеток допускається позначення додатків арабськими цифрами.

Якщо у курсовій роботі один додаток, то він позначається «Додаток А».

Кожний додаток слід починати з нової сторінки із зазначенням угорі в середині сторінки слова «Додаток» і його позначення, а під ним у дужках для обов'язкового додатка друкують слово «обов'язковий», а для інформаційного – «рекомендований» чи «довідковий». Додаток має заголовок, який друкують симетрично до тексту з великої літери окремим рядком.

Текст кожного додатка за потреби може бути поділений на розділи, підрозділи, пункти, підпункти.

Запозичена з літературних чи статистичних джерел інформація (формули, таблиці, схеми, графіки, висновки тощо) потребує обов'язкових посилань (у квадратних дужках) на порядковий номер джерела у списку використаних джерел та номери сторінок.

Список використаних джерел наводиться у такій послідовності:

- 1) Конституція України;
- 2) Кодекси законів України;

- 3) Закони України;
- 4) інструкції та нормативні акти міністерств і відомств;
- 5) наукова, навчально-методична, спеціальна література, видана курсова робота українською або російською мовами;
- 6) наукова, навчально-методична, спеціальна література, видана іноземними мовами;
- 7) електронні джерела.

ЗАХИСТ КУРСОВОЇ РОБОТИ

Після завершення написання курсової роботи студент подає та реєструє роботу на кафедрі із зазначенням строку здачі у спеціальному журналі (за підписом студента).

На курсову має бути надано відгук наукового керівника, який вирішує питання про допуск студента до захисту, роблячи відповідний запис на титульному аркуші роботи.

У разі невідповідності курсової роботи вимогам даних рекомендацій керівник може не допустити студента до її захисту.

Рішення керівника щодо недопущення студента до захисту має бути затверджене на засіданні кафедри.

Якщо робота допущена до захисту, студент має ознайомитись із відгуком і підготуватись до захисту. При цьому він повинен підготувати відповіді на питання, згадані у відгуку й показати усунені недоліки.

Захист курсової роботи відбувається перед початком екзаменаційної сесії. Процедура із захисту організовує комісія у складі принаймні трьох викладачів у присутності всіх студентів групи. Захист передбачає стислий виклад студентом головних проблем дослідження (в т.ч. за допомогою презентації) та їх вирішення упродовж 10–15 хвилин (з яких на доповідь студента відводиться 5-7 хвилин), а також відповіді на запитання членів комісії.

При оцінці курсової роботи береться до уваги:

- зміст і складність роботи;
- якість виконання;
- відповідність роботи щодо її оформлення;
- набуті студентом навички, які пов'язують теоретичні знання з питаннями їх практичного застосування;
- повнота та точність відповідей на поставлені запитання.

При захисті курсової роботи (проекту) студент має продемонструвати глибокі знання з досліджуваної теми, вміти чітко

викладати власні думки, використовувати ілюстративний матеріал, аргументовано відповідати на питання.

Оцінюючи курсову роботу, комісія враховує не лише якість самої роботи, а й вміння захистити сформульовані положення та висновки і відповідність оформлення встановленим вимогам.

Курсова робота, в якій розкрито тему, прореферовано використані джерела, здійснено аналіз певної інформаційної та статистичної бази даних, сформульовано висновки без необхідного їх обґрунтування, розроблено програмний додаток, який реалізовує лише частину із поставлених задач, буде оцінена на *«задовільно»*.

Робота, в якій зроблено власну оцінку використаних джерел, самостійно розроблено повнофункціональний додаток із інтуїтивно-зрозумілим користувацьким інтерфейсом та читабельним прокоментованим програмним кодом, зроблені висновки та сформульовано пропозиції, але вони не є достатньо аргументованими, може бути оцінена на *«добре»*.

Робота, в якій зроблено власну оцінку різноманітних джерел, побудована інформаційно-логічна модель системи, блок-схема програмного додатку, самостійно розроблено повнофункціональний додаток із інтуїтивно-зрозумілим користувацьким інтерфейсом та читабельним прокоментованим програмним кодом, виконано власні дослідження, розрахунки, і на їх основі зроблено аргументовані висновки та обґрунтовані пропозиції, може бути оцінена на *«відмінно»*.

Якщо студент не захистив курсову роботу (проект), то він вважається неатестованим з даної дисципліни і не допускається до екзамену.

Захищені курсові роботи (проекти) передаються на кафедру, де зберігаються згідно з вимогами до такого виду документів.

ДОДАТКИ

Додаток А

Київський національний торговельно-економічний університет
Кафедра кібернетики та системного аналізу

*Захищено на кафедрі кібернетики
та системного аналізу*

«__»_____20__р.

з оцінкою _____

Підписи членів комісії:

КУРСОВА РОБОТА

з дисципліни

**«АЛГОРИТМІЗАЦІЯ ТА
ПРОГРАМУВАННЯ»**

на тему:

(назва теми)

Студента(-ки) факультету _____
групи _____

_____ прізвище, ім'я, по батькові (підпис)

Науковий керівник _____

Вчений ступень, звання

Прізвище, ім'я, по батькові

(підпис)

Київ 20__

ЗРАЗКИ ОФОРМЛЕННЯ БІБЛІОГРАФІЧНИХ ОПИСІВ У СПИСКУ ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Господарський кодекс України [Електронний ресурс]: Редакція від 08.12.2016. – Режим доступу: <http://zakon.rada.gov.ua/cgi-bin/laws/main.cgi?nreg=436-15>.
2. Концепція розвитку зв'язку України до 2010 року. – Режим доступу : http://www.dstszi.gov.ua/P_baza/Postanov/post2238_2.htm.
3. Chase R.V. Production and Operations Management: alife cycle Approach. – 6 th edition. / R.V. Chase, N. Aquilano. – Boston, Home Wood, Irwin, 1992.– 234 p.
4. Артеменко, В. Б. Моделювання і прогнозування економічних рядів динаміки: Навч.посібник / В. Б. Артеменко. – Л. : Вид-во Львівської комерційної академії, 2003. – 228с.
5. Розділ Fuzzy Logic Toolbox на сайті користувачів MATLAB. – Режим доступу : <http://matlab.exponenta.ru/fuzzylogic/index.asp.htm>

Завідувачу кафедри

_____ (назва кафедри)

_____ (прізвище, ініціали)

Студент(а/ки)

_____ (прізвище, ім'я, по батькові)

_____ (факультет, курс, група, форма навчання)

З А Я В А

Прошу дозволити виконати курсову роботу (проект) з
дисципліни _____ за
(вказати назву дисципліни)
темою _____.
(вказати бажану тему курсової роботи (проекту))

_____ (дата)

_____ (підпис студента)

Київський національний торговельно-економічний університет

Кафедра _____

Дисципліна _____

Напрямок/спеціальність _____

Курс _____ Група _____ Семестр _____

**ЗАВДАННЯ
на курсову роботу (проект) студента**

_____ (прізвище, ім'я, по батькові)

1. Тема курсової роботи (проекту) _____

2. План курсової роботи (проекту) _____

3. Перелік графічного матеріалу _____

4. Термін подання студентом завершеної курсової роботи (проекту)
на кафедру _____

5. Термін захисту курсової роботи (проекту) _____

6. Дата видачі завдання _____

Студент _____

(підпис)

Науковий керівник _____

(підпис)

(прізвище, ім'я, по батькові)

Завідувач кафедри _____

(підпис)

(прізвище, ім'я, по батькові)

СПИСОК РЕКОМЕНДОВАНИХ ДЖЕРЕЛ

Основний

1. Michael Dawson Python Programming for the Absolute Beginner / Michael Dawson. - Cengage Learning PTR. – 480 pages.
2. Зацерковний В. І. Алгоритмізація та програмування: навчальний посібник / В. І. Зацерковний, В. І. Гур'єв, І. В. Фірсова. – Ніжин: НДУ ім. М. Гоголя, 2013. – 302 с.
3. Клакович Л. М., С. М. Левицька Теорія алгоритмів: Навчальний посібник. — Друге видання, доповнене / Л. М. Клакович, С. М. Левицька. — Львів : Видавничий центр ЛНУ ім. Івана Франка, 2015. — 161 с.
4. Raschka S. Python Machine Learning, 1st Edition / S. Raschka. – Birmingham: Packt Publishing, 2015. – 456 p.
5. Mark L. Learning Python, 5th Edition / L. Mark – Sebastopol: O'Reilly Media, 2013. – 648 p.

Додатковий

6. Richert W. Building Machine Learning Systems with Python / W. Richert, L. P. Coelho – Birmingham: Packt Publishing, 2013. – 290 с.
7. Мельник Р. А. Програмування для Інтернету на основі Java-технології: Навч.посібник для студентів напряму «Комп'ютерні науки» / Р. А. Мельник. – Л. : Львівська політехніка, 2003. – 184с.*
8. Степанов, В. П. Основи алгоритмізації та програмування : навч.-практ. посіб. для самостійної роботи студ. / В. П. Степанов, І. П. Ковріжних. – Харків : ХНЕУ, 2007. – 208с.*

Інтернет-ресурси

9. Національна бібліотека України імені В. І. Вернадського. – Режим доступу : <http://www.nbuv.gov.ua/>
10. Сайт розробника Python [Електронний ресурс]. – Режим доступу: <https://www.python.org>.
11. Сайт про мову програмування C++ [Електронний ресурс]. – Режим доступу: <http://www.cplusplus.com/>.
12. Крос-платформна бібліотека чисельного аналізу [Електронний ресурс]. – Режим доступу: <http://alglib.net/>.
13. Сайт розробника PyQt [Електронний ресурс]. – Режим доступу: <https://www.riverbankcomputing.com>.

* курсивом зазначені джерела, наявні в бібліотеці КНТЕУ

14. Сайт підтримки бібліотеки NumPy [Електронний ресурс]. – Режим доступу : <http://www.numpy.org>.
15. Сайт підтримки бібліотеки Pandas [Електронний ресурс]. – Режим доступу : <http://www.pandas.pydata.org/>.
16. Безкоштовні онлайн курси [Електронний ресурс]. – Режим доступу : <https://prometheus.org.ua/>.
17. Суспільство програмістів [Електронний ресурс]. – Режим доступу: <https://www.dou.ua/>
18. Довідник по IDE Visual Studio [Електронний ресурс]. – Режим доступу: <https://msdn.microsoft.com>.
19. SQLBolt Learning SQL [Електронний ресурс]. – Режим доступу: <https://sqlbolt.com/>
20. Coursera. Online courses & Credentials From Top Educators [Електронний ресурс]. – Режим доступу: <https://www.coursera.org/>
21. UdeMy: Вивчайте будь-яку тему у власному темпі [Електронний ресурс]. – Режим доступу: <https://www.udemy.com>
22. DB Browser for SQLite [Електронний ресурс]. – Режим доступу: <https://sqlitebrowser.org/>
23. Пошук питань та відповідей з програмування [Електронний ресурс]. – Режим доступу: <http://qaru.site/>
24. Reddit: підрозділ «Python» [Електронний ресурс]. – Режим доступу: <https://www.reddit.com/r/Python/>
25. Reddit: підрозділ «Data Science» [Електронний ресурс]. – Режим доступу: <https://www.reddit.com/r/datascience/>
26. Kaggle: Your home for Data Science [Електронний ресурс]. – Режим доступу: <https://www.kaggle.com/>
27. Forums: Python Anywhere [Електронний ресурс]. – Режим доступу: <https://www.pythonanywhere.com>
28. Telegram APIs [Електронний ресурс]. – Режим доступу: <https://core.telegram.org/>
29. Python framework for Telegram Bot API [Електронний ресурс]. – Режим доступу: <https://github.com/nickoala/telepot>
30. Git [Електронний ресурс]. – Режим доступу: <https://git-scm.com/>
31. The world's leading software development platform · GitHub [Електронний ресурс]. – Режим доступу: <https://github.com/>
32. Bitbucket | The Git solution for professional teams [Електронний ресурс]. – Режим доступу: <https://bitbucket.org/>
33. PyCharm - The Python IDE for Professional Developers [Електронний ресурс]. – Режим доступу: <https://www.jetbrains.com/pycharm/>

34. Anaconda Python/R Distribution [Электронный ресурс]. – Режим доступа: <https://www.anaconda.com>
35. Jupyter Notebook [Электронный ресурс]. – Режим доступа: <https://jupyter.org/>
36. The Web framework for perfectionists with deadlines [Электронный ресурс]. – Режим доступа: <https://www.djangoproject.com/>
37. Matplotlib: Python plotting [Электронный ресурс]. – Режим доступа: <https://matplotlib.org/>