

Міністерство освіти і науки України
Національний університет водного господарства та
природокористування
Навчально-науковий інститут автоматичної, кібернетики та
обчислювальної техніки
Кафедра комп'ютерних технологій та економічної кібернетики

04-05-41

**МЕТОДИЧНІ ВКАЗІВКИ ТА РОБОЧА ПРОГРАМА
ПРАКТИКИ З ПРОГРАМУВАННЯ**

для здобувачів вищої освіти першого (бакалаврського) рівня
за освітньо-професійною програмою

«Інформаційні системи та технології» спеціальності 126

«Інформаційні системи та технології» денної форми навчання

Рекомендовано науково-
методичною радою
з якості ННІ АКOT
Протокол № 10 від 22.06.2020 р.

Рівне – 2020

Методичні вказівки та робоча програма практики з програмування для здобувачів вищої освіти першого (бакалаврського) рівня за освітньо-професійною програмою «Інформаційні системи та технології» спеціальності 126 «Інформаційні системи та технології» денної форми навчання [Електронне видання] / Гладка О. М., Карпович І. М., Шевченко І. М. – Рівне : НУВГП, 2020. – 18 с.

Укладачі:

Гладка О. М., канд. техн. наук, доцент кафедри комп'ютерних технологій та економічної кібернетики;

Карпович І. М., канд. фіз.-мат. наук, доцент кафедри комп'ютерних технологій та економічної кібернетики;

Шевченко І. М., старший викладач кафедри комп'ютерних технологій та економічної кібернетики.

Відповідальний за випуск:

Грицюк П. М., д-р екон. наук, професор, завідувач кафедри комп'ютерних технологій та економічної кібернетики.

Керівник групи забезпечення спеціальності

Гладка О. М.

© Карпович І. М., Гладка О. М., Шевченко І. М., 2020
© НУВГП, 2020

ЗМІСТ

ВСТУП.....	3
1. Опис навчальної дисципліни	4
2. Загальні положення	4
3. Мета і завдання практики з програмування	5
4. Програмні результати навчання	6
5. Компетентності здобувача	7
6. Програма практики з програмування	7
7. Звітність, підсумковий контроль та критерії оцінювання результатів практики	9
8. Організація та терміни проведення практики	14
9. Рекомендована література	16
10. Інформаційні ресурси	17
ДОДАТКИ	18

ВСТУП

Методичні вказівки та робоча програма навчальної дисципліни «Практика з програмування» складені відповідно до освітньо-професійної програми «Інформаційні системи та технології» та навчального плану підготовки фахівців за першим (бакалаврським) рівнем вищої освіти спеціальності 126 «Інформаційні системи та технології», що розроблені на основі Стандарту вищої освіти України від 12.12.2018 р. і затверджені Вченою радою НУВГП, протокол № 5 від 14.06.2019 р.

Вивченню дисципліни передуює отримання компетентностей з таких дисциплін, як «Програмування», «Математична логіка та теорія алгоритмів», «Крос-платформне програмування», «Архітектура обчислювальних систем та системне програмування». Знання та навички, отримані під час проходження практики з програмування, допоможуть здобувачам вищої освіти оволодіти компетентностями з дисциплін «Проектування інформаційних систем», «Технології тестування програмних продуктів»; знадобляться під час проходження виробничої і переддипломної практики, допоможуть успішно написати та захистити випускову роботу.

1. Опис навчальної дисципліни

Найменування показників	Галузь знань, спеціальність, освітньо-професійна програма рівень вищої освіти	Характеристика навчальної дисципліни
		денна форма навчання
Кількість кредитів ECTS – 3	Галузь знань – 12 “Інформаційні технології”	Нормативна
		Рік підготовки: 2-й
		Семестр: 4-й
Модулів –1	Спеціальність – 126 “Інформаційні технології”	Лекції: –
Змістових модулів – 1		Лабораторні роботи: –
Загальна кількість годин – 90	Освітньо-професійна програма – “Інформаційні системи та технології”	Самостійна робота: 90 год.
	Рівень вищої освіти - перший (бакалавр)	Вид контролю - залік

2. Загальні положення

Навчальна дисципліна «Практика з програмування» входить до циклу фахової підготовки здобувачів вищої освіти рівня бакалавр зі спеціальності «Інформаційні системи та технології» і є важливою складовою фундаментальної підготовки фахівців галузі комп’ютерних наук та інформаційних технологій.

Практика з програмування – це вид практичної підготовки, спрямований на розширення та закріплення теоретичних і практичних знань, отриманих студентами у процесі навчання, удосконалення навичок з програмування; розробки, відлагодження і тестування програмних продуктів та їх професійного використання; формування алгоритмічного мислення і застосування набутих компетентностей до конкретних прикладних задач.

Практика з програмування має велике значення як для подальшого навчання студентів-бакалаврів, проходження ними виробничої і переддипломної практик, виконання дипломної роботи, так і для підготовки до майбутньої професійної діяльності.

3. Мета і завдання практики з програмування

Метою практики з програмування є формування загальних і спеціальних компетентностей з розробки та професійного використання прикладного програмного забезпечення (ПЗ) і комп'ютерних технологій на основі поглибленого вивчення можливостей інтегрованого середовища *Visual Studio* та їх застосування для розв'язування конкретних прикладних задач, вироблення навичок і набуття досвіду процедурного та об'єктно-орієнтованого програмування мовою високого рівня *C#*, а також відпрацювання навичок оформлення документації до розроблених програм.

Завданнями практики з програмування є:

- розширення та закріплення теоретичних знань і практичних навичок, що отримані з дисциплін «Програмування», «Математична логіка та теорія алгоритмів», «Крос-платформне програмування», «Архітектура обчислювальних систем та системне програмування»; удосконалення знання мови програмування *C#*;
- набуття досвіду з використання сучасних інформаційних технологій та інструментальних засобів для розробки, відлагодження і тестування прикладних програмних продуктів;
- набуття компетентностей самостійної роботи і практичного застосування поширених пакетів прикладних програм, розв'язування задач з використанням обробки і аналізу даних, тестування програмного забезпечення, складання технічної документації відповідно до вимог ДСТУ.

Після проходження практики здобувач повинен **знати:**

- основні етапи процесу проектування і розв'язування задач з використанням прикладного програмного забезпечення;
- призначення, можливості і технології роботи з основними засобами аналізу даних;
- призначення, можливості і технології розробки та опису розробленої програми мовою *C#* засобами *Visual Studio*;
- принципи процедурного, структурного, об'єктно-орієнтованого та візуального програмування, базові типи даних мови *C#*; оператори управління програмою;

- прийоми роботи з покажчиками при обробці символьної інформації мовою C#; правила роботи з функціями; бібліотечними функціями для обробки символьних даних у програмах, написаних мовою C#;
- прийоми розробки, налагоджування, тестування програм у сучасних операційних системах;

вміти:

- створювати і реалізовувати типові та структурні алгоритми;
- використовувати покажчики та функції з бібліотеки для обробки символьних даних мовою C#;
- використовувати сучасне інструментальне програмне забезпечення;
- проводити початкове тестування розробленого ПЗ;
- працювати з даними, поданими у вигляді таблиць, виконувати операції з даними, проводити аналіз даних, застосовувати графічні можливості прикладного ПЗ;
- використовувати сучасне інструментальне програмне забезпечення;
- створювати презентації засобами *PowerPoint* та *Macromedia Flash*.

4. Програмні результати навчання

Відповідно до освітньо-професійної програми «Інформаційні системи та технології» результати практики з програмування:

- **використовувати** базові знання інформатики й сучасних інформаційних систем та технологій, навички програмування, технології розроблення алгоритмів і комп'ютерних програм мовами високого рівня із застосуванням об'єктно-орієнтованого програмування для розв'язання задач проектування і використання інформаційних систем та технологій;
- **аргументувати** вибір програмних та технічних засобів для створення інформаційних систем та технологій на основі аналізу їх властивостей, призначення і технічних характеристик з урахуванням вимог до системи і експлуатаційних умов; мати навички налагодження та тестування програмних і технічних засобів інформаційних систем та технологій.

5. Компетентності здобувача

Під час проходження практики з програмування у здобувача вищої освіти бакалаврського рівня за освітньо-професійною програмою «Інформаційні системи та технології» формуються наступні компетентності.

Загальні компетентності:

- здатність застосовувати знання у практичних ситуаціях;
- здатність до розуміння предметної області та професійної діяльності;
- здатність вчитися і оволодівати сучасними знаннями;
- здатність до пошуку, оброблення та узагальнення інформації з різних джерел.

Спеціальні (фахові, предметні) компетентності:

- здатність аналізувати об'єкт проектування або функціонування та його предметну область;
- здатність до проектування, розробки, налагодження та вдосконалення системного, комунікаційного та програмно-апаратного забезпечення інформаційних систем та технологій, управління ними;
- здатність проектувати, розробляти та використовувати засоби реалізації інформаційних систем, технологій та інфо-комунікацій (методичні, інформаційні, алгоритмічні, технічні, програмні та інші);
- здатність управляти та користуватися сучасними інформаційно-комунікаційними системами та технологіями (зокрема, такими, що базуються на використанні Інтернет).

6. Програма практики з програмування

Практика з програмування бакалаврської підготовки за освітньо-професійною програмою «Інформаційні системи та технології» проводиться в спеціалізованих комп'ютерних лабораторіях НУВГП та, за потреби, з використанням інформаційних засобів дистанційного навчання, таких як навчальна платформа Moodle, комунікаторів Hangouts (Google) Meet чи ZOOM, електронної пошти тощо.

Практика з програмування здійснюється у формі проектування та створення прикладних програмних застосувань, що реалізують конкретні прикладні задачі з деякої предметної області, які узгоджуються студентом і керівником практики в рамках завдання на практику.

Тематичний план практики з програмування наведено в табл. 1.

Таблиця 1.

Тематичний план практики з програмування

№	Зміст робіт	Кількість годин
1.	Проходження інструктажу з безпеки життєдіяльності	На початку практики
2.	Ознайомлення з метою та програмою практики, отримання завдання	2
3.	Специфікація програмних вимог	2
4.	Написання програмного коду з використанням масивів	12
5.	Використання покажчиків. Операції з покажчиками. Масиви покажчиків	12
6.	Використання функцій	6
7.	Обробка символічної інформації. Використання стандартних бібліотечних функцій	12
8.	Розробка алгоритму та програми для розв'язання індивідуального завдання	24
9.	Тестування програмного коду	4
10.	Аналітичні інструменти прикладного ПЗ	12
11.	Створення презентації засобами PowerPoint	4
12.	Оформлення звіту згідно з ДСТУ	Протягом практики
Разом		90

Завдання практики для окремого студента можуть бути, певною мірою, пов'язані між собою чи стосуватися однієї

предметної області, що дозволить студентіві протягом практики створити єдиний проект, в якому до кожного наступного завдання будуть додаватися нові програмні функції. Це, з одного боку, зменшить обсяг загальної роботи, а з іншого – дозволить здобувачам вищої освіти потренуватися у підтримці невеликих програмних проєктів.

Під час проходження практики з програмування здобувач вищої освіти зобов'язаний:

- виконувати чинні правила внутрішнього розпорядку університету;
- пройти інструктаж і дотримуватися правил охорони праці, техніки безпеки і виробничої санітарії;
- виконувати роботу за дорученням керівника практики, відповідати за якісне і своєчасне виконання доручень;
- вести щоденник практики за етапами її проходження;
- подати на кафедру письмовий звіт про проходження практики з програмування та виконання індивідуального завдання;
- захистити основні положення, відображені у звіті.

7. Звітність, підсумковий контроль та критерії оцінювання результатів практики

За результатами виконання індивідуальних завдань студент складає звіт та створює презентацію засобами PowerPoint. Зміст звіту з практики з рекомендованою кількістю аркушів за кожним розділом наведено в табл. 2.

Звіт починається з *титального аркуша*, зразок якого наведено в додатку 1. Цей аркуш повинен бути підписаний студентом та викладачем-керівником.

У списку використаної *літератури* вказуються ДСТУ, які було використано при виконанні завдань практики (формуванні технічної документації з опису розроблених програмних застосувань) і оформленні звіту та бібліографічного опису, літературні і електронні джерела, в яких розкриваються питання предметної галузі, що аналізується. Список використаної літератури складається відповідно до стандарту [1].

Скорочення, умовні позначення, символи, одиниці та терміни за умови, що в роботі їх більше, ніж 20, та кожне з них

зустрічається в звіті більше, ніж 3 рази, повинні бути подані у вигляді окремого списку на початку звіту. В іншому випадку (якщо вони зустрічаються 3 і менше разів або їх є менше, ніж 20) – згадані елементи у вигляді списку не подаються, а розшифровуються безпосередньо в тексті звіту при першому згадуванні.

При першому згадуванні в тексті маловідомих *іншомовних термінів* їх пишуть як в українській транскрипції, так і мовою оригіналу (в дужках).

У *додатки* включають матеріали, що не ввійшли до основної частини звіту. Зокрема, у додаток можна включити фрагменти коду програм, таблиці, скрін-шоти результатів обчислень тощо. Якщо в роботі є 2 чи більше додатків, то їх нумерують послідовно, наприклад, ДОДАТОК 1, ДОДАТОК 2, ...

Таблиця 2.

Зміст звіту навчальної практики з програмування

Розділ	Кількість сторінок
Титульний аркуш	1
Зміст	1
Вступ	1
Завдання	5
Коди програм	13-15
Результати роботи програм	2-3
Результати виконання індивідуального завдання. Опис архітектури програми – специфікація програмних вимог: формулювання завдання; склад, структура модулів, зв'язки між ними; специфікація даних; план тестування	7-10
Документи, створені в результаті аналітичного дослідження	3-5
Висновки	1
Список літератури	1
Додатки	1-3

Звіт друкується на комп'ютері в текстовому редакторі Word українською мовою (або пишеться від руки чітким, розбірливим

почерком) на аркушах паперу розміром А4. **Основний текст** набирається шрифтом Times New Roman 12 або 14, стилем Звичайний. Вирівнювання абзацу за шириною (якщо інше не передбачено змістом). Відступ зліва першого рядка абзацу – 1 см. Поля: ліве – не менше, ніж 25 мм, праве – не менше, ніж 10 мм, верхнє, нижнє – не менше, ніж 15 мм.

Текст фрагментів **кодів програм** набирається шрифтом Courier New 11, напівжирним стилем з вирівнюванням ліворуч (у випадку, якщо фрагменти тексту програми наведено у додатку, розмір шрифту може бути, за потреби, зменшений до 9).

Формули набираються в редакторі MathType або Equation Editor (Текст (Text) – Times New Roman; Функції (Function) – Times New Roman; Змінні (Variable) – Times New Roman, курсив; Грецькі символи (Greek) – Symbol, курсив; Символи (Symbol) – Symbol; Вектори та матриці (Vector-Matrix) – Times New Roman, напівжирний; Числа (Number) – Times New Roman; Основний розмір (Full) – 12 pt; Індeksi першого рівня (Subscript/Superscript) – 70 % від основного розміру; Індeksi другого рівня (Sub-Subscript/Superscript) – 45 % від основного розміру; Розмір символів (Symbol) – 150 % від основного розміру; Символи-індекси (Sub-symbol) – 100 % від основного розміру).

Ілюстрації (рисунок, графіка, схеми, діаграми) слід розмішувати у звіті безпосередньо після тексту, в якому вони згадуються, або на наступній сторінці. Всі ілюстрації необхідно нумерувати, на них повинні бути посилання у звіті. За необхідності під ілюстрацією розмішують пояснювальний текст. Під ілюстрацією пишуть слово «Рис.» з номером арабськими цифрами, порядок нумерації зберігається у всьому звіті, починаючи з 1.

Таблиці виконуються у текстовому редакторі за допомогою меню «Таблиця». Таблиці слід розмішувати у звіті безпосередньо після тексту, в якому вони згадуються, або на наступній сторінці. На таблиці повинні бути посилання у звіті. Таблиці повинні мати назви, які розмішуються над таблицею. Якщо таблиць у звіті більше, ніж одна, то їх потрібно нумерувати порядковою нумерацією в межах усього звіту, починаючи з 1. Номер таблиці розмішують у правому верхньому куті над заголовком таблиці після слова «Таблиця».

Сторінки звіту слід нумерувати арабськими цифрами, підтримуючи наскрізну нумерацію усього тексту звіту. Титульний аркуш включають в загальну нумерацію сторінок, але номер на ньому не ставлять. Додатки, список літератури, а також ілюстрації та таблиці, що розміщені на окремих сторінках, включають до загальної нумерації сторінок.

Атестація за підсумками практики проводиться на підставі захисту результатів, отриманих у ході практики з програмування.

Залік з практики вноситься до заліково-екзаменаційної відомості і в залікову книжку за підписом керівника практики та прирівнюється до оцінок (заліків) із теоретичного навчання і враховується під час підведення підсумків загальної успішності студента.

Звіти студентів зберігаються на кафедрі до завершення їх навчання в університеті.

Критерії оцінювання результатів практики наведено в табл. 3.

Таблиця 3.

Оцінювання результатів практики

Зміст роботи, яка оцінюється	Кількість балів
1. Теоретична підготовка: – знання предмету; – володіння матеріалом.	15
2. Особисті характеристики: – дисциплінованість під час проходження практики; – ініціативність; – самостійність; – професійна спрямованість.	10
3. Оцінювання індивідуального завдання: – якість виконання; – якість оформлення (моделі, схеми, алгоритми)	25
4. Оформлення звіту	20
5. Своєчасність подачі звітної документації	10
6. Захист практики	20
Загальна сума балів	100

Критерії оцінювання результатів практики (у % від кількості балів, виділених на завдання, із заокругленням до цілого числа):

0% – завдання не виконано;

40% – завдання виконано частково та містить суттєві помилки методичного або розрахункового характеру;

60% – завдання виконано повністю, але містить суттєві помилки у розрахунках або в методиці;

80% – завдання виконано повністю і вчасно, проте містить окремі несуттєві недоліки;

100% – завдання виконано правильно, вчасно і без зауважень.

Шкалу оцінювання результатів практики наведено в табл. 4.

Таблиця 4.

Шкала оцінювання

Сума балів за всі види навчальної діяльності	Оцінка за національною шкалою
	для заліку
90–100	зараховано
82–89	
74–81	
64–73	
60–63	
35–59	не зараховано
0–34	

Студенти, які не виконали вимоги програми практики або отримали незадовільну оцінку, відраховуються у встановленому порядку як такі, що мають академічну заборгованість. Якщо програма практики не виконана студентом з поважної причини, то студенту надається можливість пройти практику повторно при виконанні умов, визначених університетом.

8. Організація та терміни проведення практики

Організація практики на всіх етапах спрямована на забезпечення неперервності і послідовності оволодіння студентами навичками та вміннями професійної діяльності відповідно до вимог підготовки бакалавра.

Практика проводиться відповідно до індивідуальної програми практики з програмування, узгодженої студентом та викладачем-керівником на основі загальних підходів до її змісту та структури.

Перед початком практики проводяться консультаційні збори, на яких видається завдання для проходження практики з програмування, надається вся необхідна інформація з порядку проходження практики та консультація з безпеки життєдіяльності. За результатами зборів студент заповнює щоденник, в який вносить такі дані: відомості про себе, вид практики, період проходження практики, календарний графік із переліком запланованих до виконання робіт. Календарний графік завіряється підписом керівника практики.

В перший день студент-практикант проходить інструктаж із безпеки життєдіяльності, охорони праці та протипожежної безпеки в комп'ютерній лабораторії НУВГП, де відбуватиметься практика, і робить відповідний запис у журналі.

Після закінчення практики студенти оформляють всю необхідну документацію відповідно до вимог програми практики.

Загальне методичне керівництво практикою здійснюється випусковою кафедрою. Керівництво практикою з програмування здійснює викладач – керівник практики. Керівник практики надає здобувачу вищої освіти організаційне сприяння та методичну і консультаційну допомогу у вирішенні завдань програми практики.

Керівник практики:

- погоджує програму практики з програмування;
- надає консультації студентам та проводить перевірку проходження практики студентами;
- розробляє тематику індивідуальних завдань;
- несе відповідальність за дотримання студентами правил безпеки життєдіяльності;
- здійснює контроль дотримання термінів практики та її змісту;

- надає методичну допомогу студентам під час виконання ними індивідуальних завдань;
- оцінює результати виконання програми практики студентами.

Студент-практикант:

- виконує завдання за програмою практики згідно із календарним планом практики;
- отримує від керівника практики вказівки, рекомендації та роз'яснення з усіх питань, пов'язаних з організацією та проходженням практики;
- звітує про виконану роботу відповідно до встановленого графіка.

9. Рекомендована література

1. Національний стандарт України ДСТУ 8302:2015 «Інформація та документація. Бібліографічне посилання. Загальні положення та правила складання»
2. Лук'янова В. В. Комп'ютерний аналіз даних : посібник. К. : Академія, 2003. 344 с.
3. Крушельницька О. В. Методологія та організація наукових досліджень : навч. посіб. для вищ. навч. закладів. К. : Кондор, 2003. 190 с.
4. Васильев А. С#. Объектно-ориентированное программирование М. : Питер, 2012. 320 с.
5. Бабенко Л. П., Лавріщева К. М. Основи програмної інженерії : навч. посіб. К. : Знання, 2001.
6. Албахари Д. С# 5.0. Карманный справочник. М. : Диалектика / Вильямс, 2013. 951 с.
7. Калянов Г. Н. CASE. Методы и средства системного структурного анализа и проектирования. М. : МГУ, 2000.
8. Вендров А. М. CASE-технологии. Современные методы и средства проектирования информационных систем. М. : Финансы и статистика, 1998.
9. Марченко А. Л. Основы программирования на С# 2.0. М. : Интернет-университет информационных технологий, Бином. Лаборатория знаний, 2007. 552 с.
10. Павленко Л. А. Корпоративні інформаційні системи : навч. посібник. Харків : ВД "ІНЖЕК", 2005. 260 с.
11. Таненбаум Э, Узеролл Д. Компьютерные сети. СПб. : Питер, 2012. 960 с.
12. Ляшенко І. М., Коробова М. В., Столяр А. М. Основи математичного моделювання економічних, екологічних та соціальних процесів : навч. посіб. Тернопіль: Навч. книга – Богдан, 2006. 304 с.
13. Ларман К. Применение UML 2.0 и шаблонов проектирования. Введение в объектно-ориентированный анализ и проектирование. 3-е изд. СПб. : Вильямс, 2012. 736 с.
14. Матвеева Л. Є., Волков В. А. Процес розробки програмного забезпечення. Від теорії до практики. К. : Інформаційні програмні системи, 2008. 117 с.

15. Рудаков А. В. Технология разработки программных продуктов : уч. пособие. М.: Академия, 2006. 208 с.
16. Скит Д. С#. Программирование для профессионалов. М. : Вильямс, 2011. 544 с.
17. Сфер Ч., Дэвис С. Р. С# 2008 для "чайников". М. : Вильямс, 2009. 592 с.
18. Троелсен Э. Язык программирования С# 5.0 и платформа .NET 4.5. М. : Диалектика / Вильямс, 2015. 409 с.

10. Інформаційні ресурси

1. Стандарт вищої освіти України: перший (бакалаврський) рівень, галузь знань 12 – Інформаційні технології, спеціальність 126 – Інформаційні системи та технології. (Затверджено і введено в дію наказом Міністерства освіти і науки України від 12.12.2018 № 1380.)
2. Турбал Ю. В., Гладка О. М., Степанченко О. М. (2019) Освітньо-професійна програма "Інформаційні системи та технології" першого рівня вищої освіти за спеціальністю 126 "Інформаційні системи та технології" галузі знань 12 "Інформаційні технології" URL: <http://ep3.nuwm.edu.ua/id/eprint/16961>
3. Національна бібліотека ім. В. І. Вернадського. URL: <http://www.nbuv.gov.ua/e-resources/>,
<http://www.nbuv.gov.ua/webnavigator/>
4. Рівненська обласна універсальна наукова бібліотека (м. Рівне, майдан Короленка, 6). URL: <http://www.lib.rv.ua/>
5. Рівненська централізована бібліотечна система (м. Рівне, вул. Київська, 44). URL: <http://cbs.rv.ua/>
6. Наукова бібліотека НУВГП (м. Рівне, вул. Олекси Новака, 75). URL: <http://nuwm.edu.ua/naukova-biblioteka/>,
http://nuwm.edu.ua/MySQL/page_lib.php
7. Цифровий репозиторій НУВГП. URL: <http://ep3.nuwm.edu.ua>.
8. Express API 4.x. URL: <http://expressjs.com/ru/4x/api.html> [Цит. 25.03.2020].
9. Docs NPM. URL: <https://docs.npmjs.com/all> [Цит. 25.03.2020].
10. Node.js body parsing middleware. URL: <https://github.com/expressjs/body-parser> [Цит. 25.03.2020].
11. Multer. URL: <https://ewiggin.gitbooks.io/expressjs-middleware/content/multer.html> [Цит. 06.06.2019]

ДОДАТКИ

Додаток 1

Зразок оформлення титульної сторінки звіту

Міністерство освіти і науки України
Національний університет водного господарства та
природокористування
Навчально-науковий інститут автоматики, кібернетики та
обчислювальної техніки
Кафедра комп'ютерних технологій та економічної кібернетики

З В І Т

про проходження практики з програмування

студент(а/ки) _____

(спеціальність, курс, група,

_____ прізвище, ім'я та по-батькові)

Керівник _____

(вчений ступінь, звання, посада,

_____ прізвище та ініціали)

Рівне – 20__