

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ
“ДНІПРОВСЬКА ПОЛІТЕХНІКА”**



**ФАКУЛЬТЕТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ
Кафедра інформаційних технологій та комп'ютерної інженерії**

Нікулін С.Л.

**Програма та методичні рекомендації
щодо виконання курсової роботи
з дисципліни
“Об'єктно-орієнтоване програмування”
для студентів другого курсу навчання
спеціальності 126 “Інформаційні системи та технології”**

**Дніпро
НТУ “ДП”
2020**

Програма та методичні рекомендації щодо виконання курсової роботи з дисципліни “Об’єктно-орієнтоване програмування” для студентів другого курсу навчання спеціальності 126 “Інформаційні системи та технології” / Нікулін С.Л. – Д.: НТУ «ДП», 2020. – 16 с.

Наведено програму курсової роботи з дисципліни “Об’єктно-орієнтоване програмування”, а також методичні рекомендації щодо її організації та проведення. Розкриті компоненти, що сприяють формуванню професійних компетентностей, а також підкріпленню комплексу компетентностних характеристик студентів (знань, умінь, комунікацій, автономності та відповідальності) у процесі виконання курсової роботи.

Програма та методичні вказівки містять загальні положення та рекомендації щодо складання звіту про виконання курсової роботи.

Вказівки призначені для студентів спеціальності 126 “Інформаційні системи та технології” галузі 12 “Інформаційні технології”.

Погоджено рішенням науково-методичної комісії спеціальності 126 Інформаційні системи та технології (протокол № 7 від 27.08.2020).

ЗМІСТ

ВСТУП	4
1 МЕТА ТА ЗАДАЧІ КУРСОВОЇ РОБОТИ	5
2 ОРГАНІЗАЦІЯ ПРОВЕДЕННЯ КУРСОВОЇ РОБОТИ ТА ТЕХНІКА БЕЗПЕКИ	7
3 ВИМОГИ ДО ОФОРМЛЕННЯ ЗВІТУ	9
4 ВАРІАНТИ ЗАВДАНЬ	11
5 КРИТЕРІЇ ОЦІНЮВАННЯ РЕЗУЛЬТАТІВ	13
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	15

ВСТУП

Сучасна об'єктно-орієнтована розробка програмного забезпечення дозволяє створювати складні додатки та програмні комплекси, використовуючи крупноблоковий принцип побудови інтерфейсів користувача. Традиційно створення інтерфейсів вважалась трудомістким, але використання крупних блоків, що представляють собою готові кнопки, панелі, вікна та т.ін. значно спрощує цей процес.

Такі блоки за своєю сутністю є екземплярами класів (об'єктами), доповнені спеціальними механізмами, що дозволяють використовувати їх в візуальному режимі проектування програм (додатків). Зокрема, в Object Pascal такі блоки мають назву компонентів. Навички їх створення є обов'язковим для студентів програмістських спеціальностей, зокрема 126-ої.

Таким чином, курсова робота має на меті дві основні цілі:

- поглиблення знань та практичних навичок студентів, отриманих ними у курсі Ф5 Об'єктно-орієнтоване програмування;
- надбання вмінь створювати, відлагоджувати та використовувати компоненти для візуального програмування.

□ процесі виконання курсової роботи за базову мову взято мову програмування Object Pascal та програмне середовище Delphi, як досить прості та прозорі з точки зору об'єктно-орієнтованої розробки. Але виконання курсової роботи можливо на будь якій об'єктно-орієнтованій мові програмування, яка підтримує візуальний режим створення додатків.

Тривалість курсової роботи визначається затвердженням навчальним планом за спеціальністю 126 “Інформаційні системи та технології”. Конкретні терміни проведення курсової роботи встановлюються графіком навчального процесу, який затверджується щорічно керівництвом НТУ “Дніпровська політехніка”.

1 МЕТА ТА ЗАДАЧІ КУРСОВОЇ РОБОТИ

Основною метою курсової роботи є формування компетентностей щодо процесу об'єктно-орієнтованого програмування блоків-компонентів.

Згідно освітньо-професійної програми (ОПП) з підготовки бакалаврів спеціальності 126 “Інформаційні системи та технології”, інтегральна компетентність полягає в здатності розв'язувати складні спеціалізовані задачі та практичні проблеми в області інформаційних систем та технологій або в процесі навчання, що характеризуються комплексністю та невизначеністю умов, які потребують застосування теорій та методів інформаційних технологій.”. Програмування надає певні інструменти, які використовуються для розв'язання широкого кола задач як для потреб науки, так і виробництва.

Безпосередньо курсова робота готує студентів для розробки певної частини кваліфікаційної випускової роботи бакалавра, є базою для інших дисциплін плану навчання, а також закріплює знання та уміння студентів з дисципліни “Об'єктно-орієнтоване програмування”.

Курсова робота повинна закріпити знання студентів щодо використання принципів та практичних прийомів об'єктно-орієнтованого програмування, зокрема вміння будувати програмні інтерфейси, використовуючи власноруч розроблені візуальні компоненти.

До спеціальних, фахових компетентностей ОПП, що пов'язані з цією курсовою роботою, віднесені:

КЗ 1 Здатність до абстрактного мислення, аналізу та синтезу.

КЗ 2 Здатність застосовувати знання у практичних ситуаціях.

КЗ 3 Здатність до розуміння предметної області та професійної діяльності.

КЗ 5 Здатність вчитися і оволодівати сучасними знаннями.

КЗ 6 Здатність до пошуку, оброблення та узагальнення інформації з різних джерел.

КЗ 7 Здатність розробляти та управляти проектами.

КЗ 8 Здатність оцінювати та забезпечувати якість виконуваних робіт.

КС 1 Здатність аналізувати об'єкт проектування або функціонування та його предметну область.

КС 2 Здатність застосовувати стандарти в області інформаційних систем та технологій при розробці функціональних профілів, побудові та інтеграції систем, продуктів, сервісів і елементів інфраструктури організації.

КС 3 Здатність до проектування, розробки, налагодження та вдосконалення системного, комунікаційного та програмно-апаратного забезпечення інформаційних систем та технологій

КС 4 Здатність проектувати, розробляти та використовувати засоби реалізації інформаційних систем, технологій та інфокомунікацій (методичні, інформаційні, алгоритмічні, технічні, програмні та інші).

КС 15 Здатність до використання сучасних методів розробки програмного забезпечення, яке є складовою інформаційних систем та технологій.

Основними цілями курсової роботи з програмування в рамках спеціальності 126 “Інформаційні системи та технології”, враховуючи ОПП, є наступні.

1. Закріпити практичні навички об'єктно-орієнтованого програмування для вирішення завдань побудови інтерфейсів користувача методами їх крупноблочного проектування.
2. Виконати опис компонента, який потрібно розробити, його властивостей та методів.
3. Представити на перевірку програмний код компонента та програми, що його використовує.

2 ОРГАНІЗАЦІЯ ПРОВЕДЕННЯ КУРСОВОЇ РОБОТИ ТА ТЕХНІКА БЕЗПЕКИ

Згідно навчальних планів, затверджених керівництвом НТУ “Дніпровська політехніка”, з 2020 року курсове проектування та роботи виконуються студентом як на базі обладнання комп’ютерних класів випускової кафедри, так і на власному обладнанні за бажанням студента. Зокрема, використовуються комп’ютери на базі процесорів Intel Core i3 шостого покоління або більш новітні та потужніші з тактовою частотою процесорів 2,3 ГГц та вище і з об’ємами оперативної пам’яті 4 Гб або більше.

Для виконання поставлених задач курсової роботи студент використовує програмне забезпечення:

- операційна система (ОС) Windows версій 7 та вище;
- середовище програмування Delphi версій 6 та вище, або середовище будь-якої іншої об’єктно-орієнтованої мови, що підтримує візуальний режим проектування.

Під час дії карантинних заходів студенти виконують завдання курсової роботи у віддаленому режимі за місцем самоізоляції. При цьому керівник курсової роботи періодично спілкується зі студентами за допомогою різноманітних програм, наприклад таких, як Microsoft Teams, Microsoft Zoom, Sype або електронна пошта.

До кінця означеного терміну виконання курсової роботи, студент зобов’язаний скласти звіт, який містить всі етапи та результати створення програми згідно обраного варіанту завдання та захистити його перед викладачем. Викладач, згідно критеріїв оцінювання, зараховує студенту відповідні бали.

На початку курсу студенти обов’язково проходять інструктаж з техніки безпеки та охорони праці.

Студенти зобов’язані:

- 1) пройти індивідуальний інструктаж з техніки безпеки на робочому місці
- комп’ютерному класі кафедри;

2) дотримуватися інструкцій з охорони праці для працівників своєї професії;

3) при зміні робочого місця пройти додатковий інструктаж;

4) виконувати всі рекомендації з охорони праці, які надає викладач;

5) під час дії карантинних заходів, студент повинен дотримуватися всіх правил поведінки та безпеки, встановлених або рекомендованих Міністерством охорони здоров'я України, а також внутрішніми розпорядженнями та наказами керівництва НТУ "Дніпровська політехніка".

Оскільки основна робота студента зосереджена за робочим місцем, що обладнано комп'ютером, який під'єднаний до електричної мережі, то перш за все потрібно виконувати всі норми та правила з техніки безпеки і охорони праці, які прописані в [1].

При багаторазовому порушенні студентом правил техніки безпеки та охорони праці, питання про подальше використання техніки комп'ютерного класу розглядається керівництвом кафедри та факультету з прийняттям відповідного рішення.

3 ВИМОГИ ДО ОФОРМЛЕННЯ ЗВІТУ

Звіт оформлюється у вигляді окремого документу державною мовою, згідно загальних правил оформлення звітної документації, складається від третьої особи, містить певний варіант завдання та включає наступні елементи:

- Титульний лист
- Зміст
- Вступ
- Розділ 1. Огляд рішення завдання
- Розділ 2. Реалізація програмного рішення (включаючи програмний код, скріншоти та інструкцію користувача)
- Аналіз результатів та висновки
- Список використаних джерел

Звіт починається з титульного листа, який представлений на рис. 3.1.

Всі елементи оформлення тексту звіту (висота та розмітка, подання рисунків, формул та таблиць) відповідають вимогам оформлення, які надані в [2].

□ вступі студент надає короткий опис переваги використання мови програмування та бібліотеки компонентів, що її підтримує, інструментів та технологій, дає опис завдання, а також розписує задачі, які на його думку потрібно виконати для створення заданої розробки.

В першому розділі в окремих пунктах повинні бути розміщений літературний огляд рішення обраного завдання, з посиланням на рисунки: структурно-блочне уявлення компоненту та програми, що його використовує, обґрунтуванням використаних мов програмування. Надається опис інтерфейсу користувача компоненту.

В розділі реалізації програмного рішення студент наводить структурований за модулями програмний код реалізації компоненту та програми. В коді повинні обов'язково бути коментарі. Закінчується розділ інструкцією для користувача програми.

При опису програми у розділах 1 та 2 рекомендується користуватися матеріалами лекційного курсу “Об’єктно-орієнтоване програмування” та додаткових джерел, що наведені в відповідній робочій програмі курсу.

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ НТУ “ДНІПРОВСЬКА ПОЛІТЕХНІКА”

ФАКУЛЬТЕТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ Кафедра інформаційних технологій та комп’ютерної інженерії
КУРСОВА РОБОТА з дисципліни Об’єктно-орієнтоване програмування
Варіант №_
Виконав: студент гр. _____ (Ініціали, прізвище, підпис)
Перевірив: _____ (Ініціали, прізвище, підпис)
Дата захисту: _____
Рейтингова кількість балів: _____
Інституційна оцінка: _____
 Дніпро 2020

Рис. 3.1. Зразок титульного листа курсової роботи

□ висновках студент надає короткий опис основних результатів роботи своєї розробки та її переваги перед існуючими (якщо такі є) аналогами.

Обов'язковим є надання списку використаних джерел, на які студент робив посилання у тексті звіту.

4 ВАРІАНТИ ЗАВДАНЬ

Завдання: Створити новий компонент на основі існуючого. Студент обирає один з варіантів завдань, наданих у таблиці 4.1.

Таблиця 4.1

Завдання залежно від виданого варіанту

1. Нащадок TEdit з вбудованим таймером, який «гасив» б поле введення (робив би його затемненим) через 5 секунд після закінчення введення в нього.
2. Нащадок TEdit, який би перевіряв коректність введення дійсних чисел і автоматично обробляв би помилки користувача.
3. Нащадок TEdit, з наступними новими властивостями: а) діапазон завдання чисел (2 властивості чисельного типу); б) можливість введення тільки цілих чисел; в) можливість автоматичної заміни символів-букв на великі. Включення / відключення цих опцій має регулюватися за допомогою двох нових властивостей булевого типу.
4. Нащадок TBevel з властивістю Caption і можливістю вказівки місця розташування заголовка (зверху зліва, зверху вниз, знизу зліва, знизу праворуч, по центру і т.п.) за допомогою нового властивості перераховується типу.
5. Нащадок TLabel з можливістю відображення верхніх і нижніх індексів.
6. Нащадок TBevel з доданим заголовком (нова властивість типу strings), в якому можуть відображатися верхні і нижні індекси.
7. Нащадок TLabel з можливістю вибору різних типів рамок [не менше 5-ти] (властивість перераховується типу), а також накладення тіні (властивість булевого типу).
8. Нащадок TBevel з можливістю вибору різних типів рамок [не менше 6-ти] (властивість перераховується типу), а також накладення тіні (властивість булевого типу).
9. Нащадок TLabel з можливістю градієнтної заливки в 4-х напрямках (властивість перераховується типу) і вибору кольорової палітри [не менше 8-ми варіантів в т.ч., без заливки] (властивість перераховується типу).
10. Нащадок TGroupBox з новим властивістю alignment для заголовка (перераховується типу, для вирівнювання тексту по лівому краю, правому краю або по центру) і властивістю булевого типу, що дозволяє змінювати всі букви заголовка на заголовні.
11. Нащадок TSpeedButton з вбудованим PopupMenu, що випадає при пересуванні миші над кнопкою. Пункти меню повинні братися з властивості типу TStrings компонента-кнопки. (Виконання будь-яких дій після натискання на пункти меню не обов'язково).
12. Нащадок TLabel з вбудованим таймером, що забезпечує миготіння написи з можливістю задавати інтервал миготіння (властивість цілого типу), другий колір миготіння (властивість типу TColor) і булевих властивістю, що відключає миготіння
13. Нащадок TGroupBox з можливістю нанесення фонових малюнків, що завантажуються з файлу (за допомогою нового властивості типу TBitmap).
14. Нащадок TLabel, «вдавлюють» і міняє колір при переміщенні миші над ним. Додати властивості булевого типу для включення / відключення «вдавлення» і зміни кольору, а також властивість типу TColor для завдання кольору кнопки «під» мишкою.
15. Нащадок TBevel, «вдавлюють» і міняє колір фону при переміщенні миші над ним. Додати властивості булевого типу для включення / відключення «вдавлення» і зміни кольору, а також властивість типу TColor для завдання кольору написи «під» мишкою.
16. Нащадок TPanel, «вдавлюють» і міняє колір фону при переміщенні миші над ним. Додати властивості булевого типу для включення / відключення «вдавлення» і зміни кольору, а також властивість типу TColor для завдання кольору написи «під» мишкою.

17. Насадок TGroupBox з можливістю вибору однієї з 5 додаткових видів рамок (подвійна, потрійна і т.п.) з нового властивості перераховується типу.
18. Насадок TLabel з можливістю нанесення фонових малюнків, що завантажуються з файлу (за допомогою нового властивості типу TBitmap).
19. Насадок TSpeedButton, після натискання на який повинна запускатися програма (функція WinAPI ShellExecute), певна в новому властивості типу string. Компонент повинен перевіряти існування програми, і при необхідності дозволяти знайти її за допомогою TOpenDialog.
20. Насадок TSpeedButton, після натискання на який має зберігатися в файлі вміст Мемо, ім'я якого міститься в новому властивості типу TMemo. Компонент повинен перевіряти наявність файлу, і при необхідності видавати користувачу запит про створення нового файлу.
21. Насадок TEdit з новим властивістю alignment (перераховується типу, для вирівнювання тексту по лівому краю, правому краю або по центру) і можливістю змінювати вводяться букви на великі (необхідно створити нову властивість булевого типу, що дозволяє вмикати / вимикати цю опцію).
22. Насадок TImage з можливістю нанесення різних типів рамок [не менше 6-ти] (властивість перераховується типу), а також накладення тіні (властивість булевого типу).
23. Насадок TGroupBox, «вдавлюють» і міняє колір при переміщенні миші над ним. Додати властивості булевого типу для включення / відключення «вдавлення» і зміни кольору, а також властивість типу TColor для завдання кольору написи «під» мишкою.
24. Насадок TPanel з можливістю нанесення фонових малюнків, що завантажуються з файлу (за допомогою нового властивості типу TBitmap).
25. Насадок TBevel з можливістю нанесення фонових малюнків, що завантажуються з файлу (за допомогою нового властивості типу TBitmap).
26. Насадок TPanel з можливістю малювання на Canvas'e фонових зображень, певний в новому властивості перераховується типу.
27. Насадок TLabel з можливістю вибору однієї з 8 додаткових видів рамок (подвійна, потрійна, штрих-пунктирна і т.п.) з нового властивості перераховується типу.
28. Насадок TLabel з можливістю малювання на Canvas'e фонових зображень, певний в новому властивості перераховується типу.
29. Насадок TBevel з можливістю малювання на Canvas'e фонових зображень, певний в новому властивості перераховується типу.
30. Насадок TPanel з можливістю градієнтної заливки в 8-ми напрямках (властивість перераховується типу) вибору і кольорової палітри [не менше 8-ми варіантів в т.ч., без заливки] (властивість перераховується типу)
31. Насадок TLabel з можливістю накладання тіні від символів. Для вибору напрямку падіння тіні (вліво вгору, вліво вниз, право вгору, вниз і т.п.) використовувати нову властивість перераховується типу.
32. Насадок TBevel з можливістю градієнтної заливки в 8-ми напрямках (властивість перераховується типу) вибору і нанесенням заголовка (властивість типу string) в зазначеному місці (два нових властивості типу integer).
33. Насадок TPanel з тінню і об'ємним заголовком.
34. Насадок TLabel з вбудованим PopupMenu, що випадає при пересуванні миші над кнопкою. Пункти меню повинні братися з властивості типу TStrings компонента-кнопки і містити адреси програм, які повинні запускатися (функція WinAPI ShellExecute) після натискання на пункти меню.
35. Насадок TProgressBar, з можливостями: а) вибору напрямку руху зліва-направо або справа-наліво (додати властивість перераховується типу), і б) відображення колірної розгортки (плавної зміни кольору).
36. Насадок TGroupBox. При установці Enabled: = false всі компоненти, що знаходяться на ній, повинні ставати "погаслими". Додати властивість булевого типу, що дозволяє вмикати / вимикати це можливість.
37. Насадок TGroupBox з тінню і об'ємним заголовком.
38. Насадок TPaintBox з можливістю нанесення різних типів рамок [не менше 4-х] (властивість перераховується типу), а також накладення тіні (властивість булевого типу).

5 КРИТЕРІЇ ОЦІНЮВАННЯ РЕЗУЛЬТАТІВ

Оцінювання результатів курсової роботи студентів здійснюється за допомогою прозорих процедур, що ґрунтуються на об'єктивних критеріях відповідно до Положення університету «Про оцінювання результатів навчання здобувачів вищої освіти».

Досягнутий рівень компетентностей відносно очікуваних, що ідентифікований під час контрольних заходів, відображає реальний результат навчання студента.

Оцінювання навчальних досягнень студентів НТУ «Дніпровська політехніка» здійснюється за рейтинговою (100-бальною) та інституційною шкалами (табл. 5.1). Остання необхідна (за офіційною відсутністю національної шкали) для конвертації (переведення) оцінок мобільних студентів.

Таблиця 5.1

Шкали оцінювання навчальних досягнень студентів
НТУ «Дніпровська політехніка»

<i>Рейтингова</i>	<i>Інституційна</i>
90...100	відмінно / Excellent
74...89	добре / Good
60...73	задовільно / Satisfactory
0...59	незадовільно / Fail

Курсова робота зараховується, якщо студент отримав підсумкову оцінку не менше 60-ти балів. Нижча оцінка вважається академічною заборгованістю, що підлягає ліквідації відповідно до Положення про організацію освітнього процесу НТУ «Дніпровська політехніка».

Студент на контрольних заходах має надати звіт про виконання курсової роботи, орієнтований виключно на демонстрацію дисциплінарних результатів навчання.

Основні критерії оцінювання результату захисту звіту курсової роботи представлені в таблиці 5.2.

Таблиця 5.2

Критерії оцінювання результатів виконання курсової роботи

<i>№ з/п</i>	<i>Результат виконання</i>	<i>Рейтингова кількість балів</i>	<i>Інституційна оцінка</i>
1.	- звіт містить текст з детальним описом розробки в розділах 1 та 2, а також висновки; - компонент та програма виконують всі операції за завданням;	90...100	відмінно / Excellent
2.	- звіт містить незначні відхилення від правил оформлення; - студент при захисті дав невпевнені відповіді на запитання викладача або помилявся; - компонент та програма виконують всі операції за завданням.	74...89	добре / Good
3.	звіт містить значні відхилення від правил оформлення; - студент при захисті погано володіє описом створеної розробки; - компонент або програма не виконує всі операції за завданням, але працює; - звіт поданий до захисту несвоєчасно.	60...73	задовільно / Satisfactory
4.	- компонент не працює; - звіт про виконання відсутній; - студент не з'явився на залік.	0...59	незадовільно / Fail

Кінцева кількість балів з визначених в таблиці 5.2 діапазонів встановлюється з врахуванням якості захисту виконаного звіту та вірних відповідей на поставлені викладачем запитання, відповідності сформованим компетентностям навчання.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Вимоги щодо безпеки та захисту здоров'я працівників під час роботи з екранними пристроями. Затв. Наказом Міністерства соціальної політики України 14.02.2018. Офіційний вісник України від 18.05.2018 – 2018 р., No 38, стор. 121, стаття 1352, код акта 90123/2018 (URL: <https://zakon.rada.gov.ua/laws/show/z0508-18>).
2. Нікулін С.Л. Конспект лекцій з дисципліни “Об’єктно-орієнтоване програмування” для студентів спеціальності 126 «Інформаційні системи та технології», 2020. Дистанційний курс Moodle. URL: <https://do.nmu.org.ua/course/view.php?id=3540> (дата звернення: 26.08.2020).
3. Grady Booch, Robert Maksimchuk & oth. Oriented Analysis and Design with Applications. 2007. Addison-Wesley. 720 p
4. Решевська К. С., Лісняк А. О., Борю С. Ю. Об’єктно-орієнтоване програмування : навчальний посібник для здобувачів ступеня вищої освіти бакалавра спеціальності "Комп'ютерні науки" освітньо-професійної програми "Комп'ютерні науки". Запоріжжя : ЗНУ, 2020. 94 с.
5. Кравець П.О. Об’єктно-орієнтоване програмування. Навчальний посібник. Львів: Видавництво Львівської політехніки, 2012. 624 с.
6. Львов М.С., Співаковський О.В. Вступ до об’єктно-орієнтоване програмування. Навчальний посібник. - Херсон: ХГПУ, 2000.- 238 с.
7. Антони С. Освой самостоятельно объектно-ориентированное программирование за 21 день. - М: Вильямс, 2001. - 672 с.
8. Культин Н.Б. Основы программирования в Delphi 7. СПб.: БХВ-Петербург, 2009. – 640 с.
9. Рэй Л. Создание оригинальных компонент в Delphi, – К.: НИПФ “ДиаСофт Лтд.”, 1996. – 216 с.
10. Баас Р., Фервай М., Гюнтер Х.; Delphi 4: полное руководство: пер. с нем. – К.: Издательская группа BHV, 1998. – 412 с.
11. Баженова И.Ю. Delphi 7. Самоучитель программиста. – М: КУДИЦ-ОБРАЗ, 2003. – 448 с.

Навчальне видання

Нікулін С.Л.

ПРОГРАМА ТА МЕТОДИЧНІ РЕКОМЕНДАЦІЇ
щодо виконання курсової роботи
з дисципліни
“Об’єктно-орієнтоване програмування”
для студентів другого курсу навчання
спеціальності 126 “Інформаційні системи та технології”

Електронний ресурс

Видано

□ Національному технічному університеті
«Дніпровська політехніка».

Свідоцтво про внесення до Державного реєстру ДК №1842 від 11.06.2004.
49005, м. Дніпро, просп. Дмитра Яворницького, 19.