

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

ЛЬВІВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ПРИРОДОКОРИСТУВАННЯ

Факультет механіки, енергетики та інформаційних технологій
Кафедра Інформаційних технологій



РОБОЧА ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

НАВЧАЛЬНА ПРАКТИКА (1 курс)

(назва навчальної дисципліни)

ОПП «Комп'ютерні науки»

спеціальність: 122 «Комп'ютерні науки»

першого (бакалаврського) рівня вищої освіти


Львів 2024 р.

Робоча програма навчальної дисципліни «**Навчальна практика (1 курс)**» для студентів спеціальності **122 «Комп'ютерні науки»**, перший (бакалаврський) рівень вищої освіти.

Розробники: старший викладач Заплатинський Н.Б.

Робочу програму схвалено на засіданні кафедри **Інформаційних технологій** протокол №1 від 12 серпня 2024 року.


Завідувач кафедри **Інформаційних технологій**


(підпис)

(Тригуба А.М.)
(прізвище та ініціали)

Робочу програму схвалено на засіданні методичної комісії факультету механіки, енергетики та інформаційних технологій Протокол № 1 від 29 серпня 2024 року.

Голова методичної комісії факультету механіки, енергетики та інформаційних технологій.


(підпис)

(Ковалишин С.Й.)
(прізвище та ініціали)

Вступ

Навчальна практика проводиться безпосередньо на території університету. Вона направлена на закріплення теоретичних і практичних знань, пов'язаних з набуттям ОС «Бакалавр» з комп'ютерних наук.

1. Мета навчальної практики

Отримання студентами практичних навичок роботи з сучасними інформаційно-технічними засобами, знайомство з технологіями розробки, упровадження та експлуатації програмного та інформаційного забезпечення, закріплення та поглиблення теоретичних та практичних знань отриманих студентами при вивченні спеціальних дисциплін, набуття навиків для самостійного написання програм відповідної структури.

2. Завдання навчальної практики

Основними завданнями освітньої компоненти «Навчальна практика (1 курс)» є: основні поняття програмування, принципи розробки програми, типові алгоритмічні конструкції, основні структури даних; основи структурного та функційного програмування; основи об'єктно-орієнтованого програмування. В результаті вивчення дисципліни студенти розвивають уміння аналізувати поставлену задачу, обирати доцільний метод її розв'язання; складати алгоритм вирішення задачі; писати та відлагоджувати програми, володіти сучасними засобами розроблення програмного забезпечення.

Після завершення цього курсу студент буде :

Знати:

- головні принципи програмування мовою C++;
- проектування та застосування типів даних, структур даних, списків і кортежів, словників і множин, вказівники повторення, функції, модульні пакети, файли;
- зміст і можливості технології об'єктно-орієнтованого програмування, об'єкти та класи;
- практичну і теоретичну методику оцінювання складності алгоритмів і програм;
- процедури тестування і налагодження програм;

Вміти:

- застосовувати типів даних, структур даних, списків і кортежів, словників і множин, вказівники повторення, функції, модульні пакети, файли до розв'язування задач;
- проектувати модулі та функції загального призначення;
- проектувати класи і використовувати технологію об'єктно-орієнтованого програмування;
- оцінювати складність алгоритмів і програм;
- реалізовувати класичні алгоритми мовою C++;
- тестувати програми; застосовувати системні та програмовані методи налагодження програм.

3. Зміст практики

1. Організаційний етап.

1.1. Проведення загальних організаційних зборів практикантів та первинного інструктажу з техніки безпеки.

1.2. Узгодження, видача завдань та календарних термінів проходження практики.

2. Практичний етап.

2.1. Ознайомлення із обладнанням, документацією, режимами виконання завдання.

2.2. Виконання індивідуального завдання.

3. Завершальний етап.

- 3.1. Обговорення результатів проведення Навчальної практики.
- 3.2. Підготовка звіту і щоденника за результатами проходження Навчальної практики.
- 3.3. Представлення та корегування розробленого звіту.
- 3.4. Захист звіту із проходження практики.

Зміст індивідуальних завдань конкретизується під час проходження Навчальної практики, зібрані матеріали мають бути використані для підготовки рефератів, доповідей, курсових та дипломних проектів.

Індивідуальні завдання повинні бути орієнтовані на аналіз роботи та вивчення особливостей мов програмування, фреймворків, інформаційних систем тощо.

Індивідуальні завдання видаються керівником практики перед її початком та оформлюються окремою сторінкою у звіті з проходження практики за підписом керівника.

Тривалість практики – 4 тижні.

4. Перелік тем

№	Теми
Програми мовою C++	
1.	Реалізація обчислень за математичними формулами (послідовні алгоритми), наочне введення, виведення
2.	Програмна реалізація алгоритмів з розгалуженнями.
3.	Побудова арифметичних й ітераційних циклів (задачі цілочислової арифметики, опрацювання послідовностей). –
4.	Побудова, трансформація одно- та багатовимірних масивів, у тому числі динамічних
5.	Опрацювання рядків
6.	Визначення та використання простих функцій. Рекурсія.
7.	Визначення та використання функцій вищого порядку.
8.	Визначення та використання простих структур, перевизначення операторів.
9.	Використання файлів різних форматів для зберігання та завантаження даних.
10.	Опрацювання виняткових ситуацій
11.	Моделювання зв'язних структур: одно- та двозв'язних списків, бінарних дерев, дерев пошуку.
12.	Створення найпростіших класів і об'єктів.
13.	Розробка ієрархії класів. Просте наслідування класів, створення поліморфного масиву екземплярів.
14.	Побудова класів-комполітів
15.	Розробка власного класу-контейнера з багатьма функціональними можливостями
16.	Випробування множинного, закритого, захищеного наслідування.
17.	Розробка та використання шаблону функції
18.	Розробка та використання шаблону класу
19.	Використання послідовних контейнерів і алгоритмів бібліотеки STL для створення/опрацювання колекції об'єктів
20.	Використання асоціативних контейнерів і алгоритмів бібліотеки STL для створення/опрацювання колекції об'єктів
21.	Оголошення та використання класів характеристик для налаштування шаблону функції/класу
22.	Серіалізація об'єктів
23.	Десеріалізація об'єктів

5. Порядок оформлення і захисту звіту про проходження навчальної практики

Після закінчення практики студент зобов'язаний скласти звіт про проходження навчальної практики та представити реферативну роботу, згідно отриманого індивідуального завдання.

Звіт про проходження навчальної практики оформляється на стандартних аркушах формату А4. У звіт включається: титульна сторінка, детальний опис виконуваних робіт, задач, а також фотокопії робочого вікна (скріншоти) із результатом роботи програми тощо.

Після представлення студентом звіту про проходження навчальної практики викладач - керівник практики повинен в 5-ти денний термін перевірити звіт, написати рецензію, вказати недоліки, що мають місце, оцінити звіт за відповідною шкалою.

Звіт з навчальної практики захищається студентом в 10-ти денний термін після закінчення практики і оцінюється викладачем який керує практикою.

За результатами проходження навчальної практики та захисту звіту виставляється кількість балів та оцінка за кредитно-модульною системою.

6. Очікувані результати навчання

У результаті засвоєння окремих тем із дисципліни «Навчальна практика (2 курс)» здобувачі першого (бакалаврського) рівня вищої освіти набувають знання, уміння та компетентності, що відповідають вимогам ОП «Комп'ютерні науки» спеціальності 122 «Комп'ютерні науки»:

Індекс в матриці ОП	Програмні компоненти
ЗК2	Здатність застосовувати знання у практичних ситуаціях.
ЗК3	Знання та розуміння предметної області та розуміння професійної діяльності
ЗК6	Здатність вчитися й оволодівати сучасними знаннями.
ЗК11	Здатність приймати обґрунтовані рішення.
СК3	Здатність до логічного мислення, побудови логічних висновків, використання формальних мов і моделей алгоритмічних обчислень, проектування, розроблення й аналізу алгоритмів, оцінювання їх ефективності та складності, розв'язності та нерозв'язності алгоритмічних проблем для адекватного моделювання предметних областей і створення програмних та інформаційних систем.
ПРН5	Проектувати, розробляти та аналізувати алгоритми розв'язання обчислювальних та логічних задач, оцінювати ефективність та складність алгоритмів на основі застосування формальних моделей алгоритмів та обчислюваних функцій.
ПРН9	Розробляти програмні моделі предметних середовищ, вибирати парадигму програмування з позицій зручності та якості застосування для реалізації методів та алгоритмів розв'язання задач в галузі комп'ютерних наук.

7. Рекомендована література

1. Ярошко С. А. Методи розробки алгоритмів. Програмування мовою C++: навч. посібник / С.А. Ярошко, О.С. Ярошко – Львів: ЛНУ імені Івана Франка, 2022. – 248 с. – <https://lnuittutor.github.io/>
2. Бублик В.В. Об'єктно-орієнтоване програмування: [Підручник] / В.В. Бублик. – К.: ІТ-книга, 2015. – 624 с.
3. Stephen Prata C++ Primer Plus 6th Edition (Developer's Library) In 2 Volumes – Addison-Wesley

Professional, 2011. – 1440 p.

4. Дудзяний І.М. Програмування мовою С++. Частина 1: Парадигма процедурного програмування: навчальний посібник / І.М. Дудзяний. – Львів: ЛНУ імені Івана Франка, 2013. – 468 с.

5. Microsoft Learn: С++ language documentation. – Електронний ресурс. Режим доступу: <https://learn.microsoft.com/en-us/cpp/cpp/?view=msvc-170>

6. <https://git-scm.com/book/uk/v2> Додаткова література

7. Bruce Eckel Thinking in C++, Vol. 1: Introduction to Standard C++, 2nd Edition – Prentice Hall, 2000. – 814 p.

8. Bruce Eckel Thinking in C++, Vol. 2: Practical Programming, 2nd Edition – Prentice Hall, 2003. – 832 p.

9. Scott Meyers Effective Modern C++ – O'Reilly Media, 2015. – 316 p.

10. Nicolai M. Josuttis The C++ Standard Library

Інформаційні ресурси

1. Бібліотечно-інформаційні ресурси – книжковий фонд, періодика та фонди на електронних носіях бібліотеки ЛНУП, державних органів науково-технічної інформації, наукових, науково-технічних бібліотек та інших наукових бібліотек України.

2. Електронний ресурс стосовно ІТ компаній Львівського регіону: http://www.invest-lvivregion.com/it-компанії_ua_285cms.htm

3. Портал об'єктно-орієнтованого програмування: <http://oop.in.ua/tag/FAQ/>

4. Книжки з програмування: як читати і що саме: <https://dou.ua/lenta/articles/programming-books/>

5. Українська технічна література. Програмування: <https://ukrtechlibrary.wordpress.com/tag/програмування/>

6. Задачі програмування із прикладами розв'язку: <http://purecodecpp.com/uk/archives/433>

7. Задачі програмування із прикладами розв'язку: <http://library.nuft.edu.ua/ebook/datathree.php?ID=138>

8. Задачі програмування із прикладами розв'язку: <http://abramov.org.ua/blog/category/opp/obchislennya-%D1%96z-zber%D1%96gannyam-posl%D1%96dovnosti/>

9. Електронне навчання у віртуальному навчальному середовищі ЛНУП (<https://moodle.lnup.edu.ua/>).

Політика оцінювання

Політика щодо дедлайнів та перескладання: Звіти, які здаються із порушенням термінів без поважних причин, оцінюються на нижчу оцінку (75% від можливої максимальної кількості балів за вид діяльності балів). Перескладання звіту із Навчальної практики відбувається за поважних причин (наприклад, лікарняний).

Політика щодо академічної доброчесності: Списування під час виконання завдань заборонені (в т.ч. із використанням мобільних девайсів). Мобільні пристрої дозволяється використовувати лише під час он-лайн тестування та підготовки практичних завдань під час практики.

Політика щодо відвідування: Відвідування занять є обов'язковим компонентом оцінювання. За об'єктивних причин (наприклад, хвороба, працевлаштування, міжнародне стажування) навчання може відбуватись в он-лайн формі за погодженням із керівником курсу.