



ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ

Робоча програма навчальної дисципліни (Силабус)

Реквізити навчальної дисципліни

Рівень вищої освіти	<i>Перший</i> (бакалаврський)
Галузь знань	16 Хімічна інженерія та біоінженерія
Спеціальність	162 Біотехнології та біоінженерія
Освітня програма	<i>Біотехнологія</i>
Статус дисципліни	Нормативна
Форма навчання	Денна
Рік підготовки, семестр	2 курс, осінній семестр
Обсяг дисципліни	<i>4 кредити, 36 год лекції, 36 год практичні заняття, самостійна робота 48 год</i>
Семестровий контроль/ контрольні заходи	<i>Залік / модульна контрольна робота / ДКР</i>
Розклад занять	https://schedule.kpi.ua
Мова викладання	Українська
Інформація про керівника курсу / викладачів	Лектор: <i>д.т.н., проф., проф. Горобець Світлана Василівна</i> Практичні д. т. н., проф., проф. Горобець Світлана Василівна Листування зі студентами з організаційних питань буде здійснюватися через WhatsApp чат дисципліни
Розміщення курсу	На платформі дистанційного навчання Google Клас за посиланням https://classroom.google.com/c/NDA4MTY0OTYzOTA3?cjc=3ku5mbm код курсу 3ku5mbm, постійне посилання на Google Meet https://meet.google.com/uom-mkaz-tem

Програма навчальної дисципліни

1. Опис навчальної дисципліни, її мета, предмет вивчення та результати навчання

Сучасний світ вже не можливо уявити без інформаційних технологій, які увійшли у всі галузі. Професійний біотехнолог наразі має знати та вміти використовувати сучасні мови програмування. Лідером серед таких мов є Python та Biopython, що є колекцією некомерційних інструментів для обчислення в біології та біоінформатиці з відкритим кодом створеною міжнародною асоціацією розробників Python – це набір модулів Python, які надають функції для роботи з послідовностями ДНК, РНК і білкових послідовностей, такими як зворотна комплементация ланцюжка ДНК, пошук мотивів в білкових послідовностях тощо. Він надає безліч синтаксичних аналізаторів для читання всіх основних генетичних баз даних, таких як GenBank, SwissPort, PDB, а також оболонки / інтерфейси для запуску інших популярних програм / інструментів біоінформатики, таких як NCBI BLASTN, Entrez тощо, в середовищі Python. Курс знайомить студентів з основами програмування на мові Python та з низкою пакетів мови програмування Python для аналізу експериментальних даних в біотехнології: пакетом Matplotlib для побудови графіків та діаграм, пакетами Python для задач регресійного аналізу, пакетом Opensv-python для аналізу зображень тощо. Даний курс знайомить майбутніх біотехнологів з основами біоінформатики, що яка є лідером в розробці лікарських препаратів, знайомить з програмними засобами, що використовуються для аналізу та порівняння

генетичних даних, сучасними методами геноміки, протеоміки, фармакогеноміки та вчить отримувати нові знання за використання методів біоінформатики.

Метою навчальної дисципліни є формування у студентів наступних компетентностей:

- ЗК -01 Здатність до пошуку, оброблення та аналізу інформації з різних джерел
- ФК-06: Здатність застосовувати сучасні методи та інструменти досліджень, і цифрові технології, бази даних та інші електронні ресурси, спеціалізоване програмне забезпечення у науковій та викладацькій діяльності.

Після засвоєння навчальної дисципліни студенти мають продемонструвати такі результати навчання:

Програмні результати навчання:

- ПРН-3: Знання і розуміння проблемних питань сучасної біотехнології (в тому числі і на межі предметних галузей) та біоінженерії для створення новітніх біотехнологій.
- ПРН-05: Мати передові концептуальні та методологічні знання з біотехнологій та біоінженерії і на межі предметних галузей, а також дослідницькі навички, достатні для проведення наукових і прикладних досліджень на рівні останніх світових досягнень з відповідного напрямку, отримання нових знань та/або здійснення інновацій.
- ПРН-07: Застосовувати сучасні інструменти і технології пошуку, оброблення та аналізу інформації, зокрема, статистичні методи аналізу даних великого обсягу та/або складної структури, спеціалізовані бази даних та інформаційні системи.

2. Пререквізити та постреквізити дисципліни (місце в структурно-логічній схемі навчання за відповідною освітньою програмою)

Місце в структурно-логічній схемі навчання забезпечується дисциплінами, такими як загально технічна дисципліна «Вища математика», а також базовий рівень володіння англійською мовою не нижче А2. У структурно-логічній площині програми підготовки бакалаврів з біотехнології дисципліна базується на попередньо вивчених дисциплінах в школі. Вивчення дисципліни впливає на подальший шлях здобувачів вищої освіти. На базі дисципліни «Інформаційні технології» в подальшому вивчають наступні дисципліни «Методи аналізу в біотехнології» та на блок дисциплін зв'язаних з проектуванням біотехнологічних виробництв.

3. Зміст навчальної дисципліни

Тема 1. Інтерпретована, об'єктно-орієнтована мова програмування Python.

Лекція 1. Різні мови програмування та сфери їх застосування. Об'єктно-орієнтований і системний підходи. Порівняння об'єктно-орієнтованого та процедурного (традиційного) підходів. Особливості мови Python. Інтегральні середовища розробки (ICP): Pycharm Community, IDLE, Spyder.

Лекція 2. Базові поняття мови Python: базовий синтаксис, типи даних, математичні операції, логічні оператори, умовні оператори, цикли.

Лекція 3. Складні структури даних мови Python. Робота з рядками, списками, словниками, кортежами, множинами (наборами).

Лекція 4. Робота з файлами в мові Python.

Лекція 5. Основи пакету numpy мови програмування Python.

Лекція 6. Функції в Python.

Лекція 7. Класи та проекти в Python.

Тема 2. Використання мови програмування Python для аналізу експериментальних даних в біотехнології.

Лекція 8. Пакет Matplotlib для побудови графіків та діаграм.

Лекція 9. Пакети Python для задач регресійного аналізу. Лінійна регресія експериментальних даних.

Лекція 10. Пакет Opensv-python для аналізу зображень, отриманих методами оптичної мікроскопії, скануючої зондової мікроскопії, скануючої електронної мікроскопії. Зчитування зображень біооб'єктів із файлів, їх запис у файл.

Тема 3. Основи біоінформатики. Базові алгоритми біоінформатики на мові Python.

Лекція 11. Формування біоінформатики як самостійної науки. Предмет і задачі біоінформатики.

Лекція 12. Основи пакету Biopython для роботи з базами даних. Попарне та множинне вирівнювання послідовностей в Biopython. Організація доступу до бази даних NCBI Entrez в пакеті Biopython.

Лекція 13. Методи динамічного програмування. Алгоритм глобального вирівнювання двох послідовностей та приклад реалізації алгоритму глобального вирівнювання двох послідовностей на мові програмування Python.

Лекція 14. Модифікації алгоритму вирівнювання генетичних послідовностей. Алгоритми локального, псевдоглобального та швидкого вирівнювання двох послідовностей.

Лекція 15. Хешування. Використання хеш-таблиць в базах даних для пошуку співпадінь по списку слів. Алгоритм побудови точкової матриці (dot-матриці).

Лекція 16. Особливості алгоритмів вирівнювання білкових послідовностей. Матриці амінокислотних замінів PAM і BLOSUM, PSSM матриці.

Лекція 17. Система премій і штрафів для вирівнювання послідовностей. Алгоритм загальної функції штрафу.

Лекція 18. Біоінформатика як інструмент отримання нових знань.

Надається перелік розділів і тем всієї дисципліни.

Навчальні матеріали та ресурси

Базова література:

1. Горобець О.Ю. Мова python: для інженерних та наукових задач [Електронний ресурс] : підручник для здобувачів ступеня бакалавра за спеціальністю 162 «Біотехнології та біоінженерія», магістра за спеціальністю 162 «Біотехнології та біоінженерія», бакалавра за спеціальністю 104 «Фізика та астрономія» та магістра за спеціальністю 104 «Фізика та астрономія» / О. Ю. Горобець, С. В. Горобець, К. Ю. Хахно ; КПІ ім. Ігоря Сікорського. – 1-ше вид. – Електрон. текст. дані (1 файл, 13.7 авт. арк) – Київ : КПІ ім. Ігоря Сікорського, 2024. – 277 с. <https://ela.kpi.ua/items/29c3b487-d8b2-4fb1-b5b6-2c2332efa518>
2. Горобець С. В. Основи біоінформатики [Електронний ресурс]: підручник для студентів напряму підготовки 6.051401 «Промислова біотехнологія» факультету біотехнології і біотехніки / С. В. Горобець, О. Ю. Горобець, Т. А. Хоменко ; НТУУ «КПІ». - Електронні текстові дані (1 файл: 2,72 Мбайт). – Київ : НТУУ «КПІ», 2010. <https://ela.kpi.ua/bitstream/123456789/774/1/10-11-146.pdf>
3. Горобець С. В. Біоінформатика. Практикум [Електронний ресурс]: навч. посіб. для студ. спеціальності 162 «Біотехнології та біоінженерія» / С. В. Горобець, О. Ю. Горобець, І.В. Дем'яненко; КПІ ім. Ігоря Сікорського. – Електронні текстові дані (1 файл 5.49 Мбайт). – Київ : КПІ ім. Ігоря Сікорського, 2020. – 86с. <https://ela.kpi.ua/handle/123456789/38813>
4. Горобець С. В. Біоінформатичні бази даних [Електронний ресурс] : навч. посіб. для студ. спеціальності 162 «Біотехнології та біоінженерія» / С. В. Горобець, О. Ю. Горобець, М. О. Булаєвська ; КПІ ім. Ігоря Сікорського. – Електронні текстові дані (1 файл: 3,86 Мбайт). – Київ : КПІ ім. Ігоря Сікорського, 2020. – 117 с. <https://ela.kpi.ua/handle/123456789/36457>

5. Васильєв О.М. Програмування в Python. Теорія і практика : навч. посіб. Київ: Видавництво Ліра-К, 2023. 462 с. <https://lira-k.com.ua/files/contents/13141.pdf>
6. Горобець С. В. Інформаційні технології. Домашня контрольна робота [Електронний ресурс]: навч. посіб. для здобувачів ступеня бакалавра за освіт. програмою «Біотехнології» спец. 162 Біотехнології та біоінженерія / КПІ ім. Ігоря Сікорського – Електрон. текст. дані (1 файл, 0.5 авт. арк). – Київ : КПІ ім. Ігоря Сікорського, 2024. – 18 с. <https://ela.kpi.ua/items/19077b33-3f71-44e6-b616-b172c03cdb6e/full>

Додаткова:

1. Костюченко А.О. Основи програмування мовою Python: навчальний посібник. Ч.: ФОП Баликіна С.М., 2020. 180 с.
<https://epub.chnpu.edu.ua/jspui/bitstream/123456789/5584/1/%D0%9E%D1%81%D0%BD%D0%BE%D0%B2%D0%B8%20%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D1%83%D0%B2%D0%B0%D0%BD%D0%BD%D1%8F%20%D0%BC%D0%BE%D0%B2%D0%BE%D1%8E%20Python.pdf>
2. Яковенко А.В. Основи програмування. Python. Частина 1 [Електронний ресурс]: підручник для студ. спеціальності 122 "Комп'ютерні науки", спеціалізації "Інформаційні технології в біології та медицині" / А. В. Яковенко ; КПІ ім. Ігоря Сікорського. – Електронні текстові дані (1 файл: 1,59 Мбайт). – Київ : КПІ ім. Ігоря Сікорського, 2018. – 195 с. <https://ela.kpi.ua/server/api/core/bitstreams/dbbe8ff5-11d7-4a92-918c-d1445c3d20a7/content>
3. Bioinformatics. - editor: Nakaya Helder I., PhD, editor Information Brisbane (AU): 2021. – 188 p. <https://www.ncbi.nlm.nih.gov/books/NBK569562/>
4. Осташ Богдан. Біоінформатика: аналіз генетичних послідовностей – Львів: ЛНУ ім. Франка, 2022.- 232 с. <http://dspace.lnlibrary.lviv.ua/jspui/handle/123456789/169>
5. Lesk A.M. Introduction to Bioinformatics. – Oxford. – 2013. – 400 p. https://openlibrary.org/books/OL28516721M/Introduction_to_Bioinformatics

Інформаційні ресурси

1. <https://www.ncbi.nlm.nih.gov/>
2. <https://www.python.org/>
3. <https://matplotlib.org/>
4. <https://scikit-learn.org/stable/>
5. https://docs.opencv.org/4.5.2/d6/d00/tutorial_py_root.html
6. <http://www.biopython.org>

Навчальний контент

4. Методика опанування навчальної дисципліни (освітнього компонента)

1	2
	<i>Тема 1. Інтерпретована, інтегрована, об'єктно-орієнтована мова програмування Python.</i>
1	Лекція 1. Різні мови програмування та сфери їх застосування. Об'єктно-орієнтований і системний підходи. Порівняння об'єктно-орієнтованого та процедурного (традиційного) підходів. Особливості мови Python. Інтегральні середовища розробки (ICP): Pycharm Community, IDLE, Spyder. Література: базова [1, 5], додаткова [1-2], інформаційні джерела [1-6]

2	Лекція 2. Базові поняття мови Python: базовий синтаксис, типи даних, математичні операції, логічні оператори, умовні оператори, цикли. Література: базова [1, 5], додаткова [1-2], інформаційні джерела [2]
3	Лекція 3. Складні структури даних мови Python. Робота з рядками, списками, словниками, кортежами, множинами (наборами). Література: базова [1, 5], додаткова [1-2], інформаційні джерела [2]
4	Лекція 4. Робота з файлами в мові Python. Література: базова [1, 5], додаткова [1-2], інформаційні джерела [2]
5.	Лекція 5. Основи пакету numpy мови програмування Python. Література: базова [1, 5, 6], додаткова [1-2], інформаційні джерела [2]
6.	Лекція 6. Функції в Python. Види функцій, аргументи функцій, документація функцій тощо. Література: базова [1, 5], додаткова [1-2], інформаційні джерела [2]
7.	Лекція 7. Класи та проекти в Python. Література: базова [1, 5], додаткова [1-2], інформаційні джерела [2]
	<i>Тема 2. Використання мови програмування Python для аналізу експериментальних даних в біотехнології.</i>
8	Лекція 8. Пакет Matplotlib для побудови графіків та діаграм. Література: базова [1, 5], інформаційні джерела [3]
9	Лекція 9. Пакети Python для задач регресійного аналізу. Лінійна регресія експериментальних даних. Література: базова [1, 5], додаткова [3], інформаційні джерела [2-5]
10	Лекція 10. Пакет Opensv-python для аналізу зображень, отриманих методами оптичної мікроскопії, скануючої зондової мікроскопії, скануючої електронної мікроскопії. Зчитування зображень біооб'єктів із файлів, їх запис у файл тощо. Література: базова [1, 5], інформаційні джерела [2,5]
<i>Тема 3. Основи біоінформатики. Базові алгоритми біоінформатики на мові Python.</i>	
11	Лекція 11. Формування біоінформатики як самостійної науки. Предмет і задачі біоінформатики. Література: базова [1-4].
12	Лекція 12. Основи пакету Biopython для роботи з базами даних. Попарне та множинне вирівнювання послідовностей в Biopython. Організація доступу до бази даних NCBI Entrez в пакеті Biopython. Література: базова [1-4], інформаційні джерела [1, 6]
13	Лекція 13. Методи динамічного програмування. Алгоритм глобального вирівнювання двох послідовностей та приклад реалізації алгоритму глобального вирівнювання двох послідовностей на мові програмування Python. Література: базова [2-4], додаткова [4, 5], інформаційні джерела [1]
14	Лекція 14. Модифікації алгоритму вирівнювання генетичних послідовностей. Алгоритми локального, псевдоглобального та швидкого вирівнювання двох послідовностей. Література: базова [2-4], інформаційні джерела [1]
15	Лекція 15. Хешування. Використання хеш-таблиць в базах даних для пошуку співпадінь по списку слів. Алгоритм побудови точкової матриці (dot-матриці). Література: базова [2-4], інформаційні джерела [1]
13	Лекція 16. Особливості алгоритмів вирівнювання білкових послідовностей. Матриці амінокислотних заміन PAM і BLOSUM, PSSM матриці. Література: базова [2-4], інформаційні джерела [1]

17	Лекція 17. Система премій і штрафів для вирівнювання послідовностей. Алгоритм загальної функції штрафу. Література: базова [2-4], інформаційні джерела [1]
18	Лекція 18. Біоінформатика як інструмент отримання нових знань. Література базова: базова [2-4], додаткова [3, 5], інформаційні джерела [1]

Практичні заняття

Основні завдання циклу практичних занять:

- розробка елементарних програм в Python
- робота з програмними пакетами, які використовуються в біотехнології;
- практична робота з сучасними біотехнологічними базами даних та їх основними алгоритмами.

1.	Практична робота 1 Знайомство та основи роботи в базах даних Національного центру біотехнологічної інформації США (National Center for Biotechnology Information (NCBI)). Робота з базами даних PMC та PubMed. Інтегральні середовища розробки (ICP): Pycharm Community, IDLE, Spyder. Література: базова [1-4], інформаційні ресурси [1].
2.	Практична робота 2. Налаштування середовища розробки та запуск програми на python. Введення та виведення даних. Математичні операції. Умовні оператори, цикли. Література: базова [1,5], додаткова [1,2] інформаційні джерела [2]
3.	Практична робота 3. Робота з типами даних мови Python: рядками, списками, словниками, кортежами, множинами (наборами). Література: базова [1,5], додаткова [1,2] інформаційні джерела [2]
4.	Практична робота 4. Робота з файлами в мові Python. Зчитування послідовностей з баз даних NCBI та запис їх у файли. Література: базова [1,5], додаткова [1,2] інформаційні джерела [2]
5.	Практична робота 5 Основи пакету numpy мови програмування Python. Функції пакету numpy, робота з масивами. Література: базова [1,5], додаткова [1,2] інформаційні джерела [2]
6.	Практична робота 6 Функції в Python, види функцій, локальні та глобальні змінні. Література: базова [1,5], додаткова [1,2] інформаційні джерела [2]
7.	Практична робота 7 Класи та проекти в Python. Література: базова [1,5], додаткова [1,2] інформаційні джерела [2]
8.	Практична робота 8 Робота з пакетами Matplotlib для побудови графіків та діаграм. Література: базова [1], інформаційні ресурси [3].
9.	Практична робота 9

	Пакети мови програмування Python для задач регресійного аналізу. Лінійна регресія експериментальних даних. Література: базова [1], додаткова [3], інформаційні ресурси [2-5].
10.	Практична робота 10 Пакет Opensv-python для аналізу зображень, отриманих методами оптичної мікроскопії, скануючої зондової мікроскопії, скануючої електронної мікроскопії. Зчитування зображень біооб'єктів із файлів, їх запис у файл. Література: базова [1], інформаційні джерела [2-5]
11.	Практична робота 11. Порівняння послідовностей з використанням програми BLAST. Програми родини BLAST. Література: базова [2-4], додаткова [4], інформаційні ресурси [1].
12.	Практична робота 12. Основи пакету Biopython для роботи з базами даних. Організація доступу до бази даних NCBI Entrez в пакеті Biopython. Література: базова [1], інформаційні джерела [6]
13.	Практична робота 13 Алгоритм глобального вирівнювання та його реалізація на мові Python. Література: базова [2-4], інформаційні джерела [2]
14.	Практична робота 14 Алгоритм локального, псевдоглобального та швидкого вирівнювання двох послідовностей та реалізація алгоритму локального вирівнювання на мові Python. Література: базова [2-4], додаткова [1,2] інформаційні джерела [2]
15.	Практична робота 15 Алгоритм хешування. Побудова хеш-таблиць. Побудови dot-матриць в NCBI. Література: базова [2-4], інформаційні ресурси [1].
16.	Практична робота 16. Особливості алгоритмів вирівнювання білкових послідовностей. Розрахунок матриці амінокислотних замінів PAM. Література: базова [2-4], інформаційні джерела [2]
17.	Практична робота 17 Особливості алгоритмів вирівнювання білкових послідовностей. Розрахунок матриці PSSM. Література: базова [2-4], інформаційні джерела [2]
18.	Практична робота 18 МКР

Самостійна робота студента

Самостійна робота студента по дисципліні включає підготовку до аудиторних занять (28 годин), модульної контрольної (4 години), написання ДКР (10 годин), підготовка до заліку (6 годин).

Політика та контроль

5. Політика навчальної дисципліни (освітнього компонента)

Вивчення дисципліни «Інформаційні технології» відбувається на лекційних та практичних заняттях. Наочність навчальних занять забезпечується використанням значної кількості ілюстративного матеріалу (схем, таблиць, слайдів). Під час викладання даної дисципліни викладач

проводить опитування здобувачів для того, щоб визначити рівень засвоєння ними викладеного матеріалу, важливим є активність здобувачів. Практичні заняття проходять з використанням комп'ютерної техніки та відповідного програмного забезпечення.

Положення про рейтингову систему оцінювання з дисципліни «Інформаційні технології» до даної робочої навчальної програми представлені в пункті 9 силабуса.

Зазначається система вимог, які викладач ставить перед студентом/аспірантом:

- *правила відвідування занять (як лекцій, так і практичних/лабораторних);*
Відвідування лекцій, практичних занять та лабораторних робіт, а також відсутність на них, не оцінюється. Однак, студентам рекомендується відвідувати заняття, оскільки на них викладається теоретичний матеріал та розвиваються навички, необхідні для формування компетентностей, визначених стандартом освіти. Система оцінювання орієнтована на отримання балів за активність студента, а також виконання завдань, які здатні розвинути практичні уміння та навички. За об'єктивних причин (наприклад, хвороба, працевлаштування, міжнародне стажування тощо) навчання може відбуватися в он-лайн формі за погодженням із керівником курсу.
- *правила поведінки на заняттях (активність, підготовка коротких доповідей чи текстів, відключення телефонів, використання засобів зв'язку для пошуку інформації на гугл-диску викладача чи в інтернеті тощо);*

На аудиторних заняттях студент має поважати викладача та дисципліну, що він слухає; Виконувати елементарні правила та норми поведінки; Протягом заняття забороняється користуватися мобільними телефонами, окрім екстрених випадків. Норми етичної поведінки студентів і працівників визначені у розділі 2 Кодексу честі Національного технічного університету України «Київський політехнічний інститут імені Ігоря Сікорського». Детальніше: <https://kpi.ua/code>.

- *правила призначення заохочувальних та штрафних балів;*
за активну роботу на лекціях та практичних заняттях студент може отримати до 10 балів сумарно протягом семестру.
- *політика дедлайнів та перескладань;*
Термін здачі кожного виду роботи обговорюється на занятті під час видачі завдання та залежить від типу роботи.

політика щодо академічної доброчесності;
визначені у розділі 3 Кодексу честі Національного технічного університету України «Київський політехнічний інститут імені Ігоря Сікорського». Детальніше: <https://kpi.ua/code>. Використання додаткових джерел інформації під час оцінювання знань заборонено (у т.ч. мобільних пристроїв). Мобільні пристрої дозволяється використовувати лише під час он-лайн тестування та виконання розрахунків.

У випадку виявлення академічної недоброчесності (плагіату в ІСЗ, списування при виконанні МКР або залікової роботи тощо) робота не зараховується.

Повторне написання МКР передбачено за розкладом консультацій.

У разі виявлення академічної недоброчесності під час виконання екзаменаційної/залікової роботи студент усувається з екзамену/заліку і може повторити спробу здачі семестрового контролю на 1 чи 2 перездачі.

У разі виявлення академічної недоброчесності під час виконання /ДКР тощо роботи (плагіат абощо) завдання студенту не зараховується. Повторне виконання індивідуального завдання можливе тільки за іншим варіантом / Виправлене індивідуальне завдання оцінюється за критеріями, як таке, що виконане невчасно

інші вимоги, що не суперечать законодавству України та нормативним документам Університету.

- Студент зобов'язаний зареєструватися на платформі дистанційного навчання Google Workspace for Education Fundamentals (в минулому G Suit For Education) на домені @LLL.kpi.ua та приєднатися до Google Класу «Інформаційні технології» за посиланням

<https://classroom.google.com/c/NDA4MTY0OTYzOTA3?cjc=3ku5mbm> . Для цього студенту необхідно спочатку отримати акаунт в Google Workspace for Education Fundamentals на домені @LLL.kpi.ua. Для отримання акаунту в Google Workspace for Education Fundamentals на домені @LLL.kpi.ua необхідно заповнити Google Форму: https://sikorsky-distance.kpi.ua/reg_gsuite/. Після реєстрації та модерації заявки студента, адміністратор надішле студенту на пошту пароль та логін до акаунту, з яким студент зможе використовувати всі доступні інструменти та сервіси Google Workspace for Education Fundamentals. Google Workspace for Education Fundamentals – це пакет спеціалізованого хмарного програмного забезпечення, інструментів для спільної роботи та дистанційного навчання від компанії Google. Основна складова пакету – система управління навчанням Google Клас, яка дозволяє викладачу створювати навчальні класи, оцінювати завдання, надавати студентам зворотній зв'язок, публікувати оголошення і поширювати навчальні матеріали. Викладач може бачити, хто виконав завдання, а хто ще продовжує над ним працювати, а також читати питання і коментарі студентів. Для приєднання до навчального курсу «Інформаційні технології» студенту потрібно перейти у Google Клас за посиланням <https://classroom.google.com>, натиснути зображення «+» у верхньому правому кутку браузера, вибрати «Приєднатися до класу» та ввести **код курсу 3ku5mbm**. Акаунти студентів, які приєдналися до Google Класу не з акаунту на домені @LLL.kpi.ua, будуть вилучатися з навчального курсу «Інформаційні технології» Google Класу тому, що автоматичний імпорт оцінок за тестування можливий виключно з акаунту Інформаційні технології на домені @LLL.kpi.ua. Система Google Клас автоматично надсилає кожному студенту бали по кожному з видів контролю на електронну пошту. Тому для ознайомлення з балами за кожен окремий вид контролю студенту необхідно змінити налаштування електронної пошти так, щоб ці електронні листи не потрапляли у спам. Всі виконані завдання для перевірки викладачем студент повинен завантажувати через систему Google Клас (результати виконання завдань, надіслані через телеграм канал перевірятися не будуть).

- Листування із студентами з організаційних питань буде здійснюватися через телеграм чат дисципліни.

6. Види контролю та рейтингова система оцінювання результатів навчання (PCO)

Поточний контроль: бали за роботу на практичних заняттях, а саме за відповіді на запитання та написання коду на мові python (всього 45 балів за семестр); тестування знань студентів з лекційного матеріалу та практичних робіт, за семестр 7 тестів по 5 балів кожний (всього 35 балів); виконання домашньої контрольної роботи (10 балів) та модульної контрольної роботи (10 балів). Загальна сума балів за семестрову роботу – 100 балів. Докладніша інформація щодо поточного контролю та критеріїв оцінювання наведена в PCO з дисципліни (Додаток 1).

Календарний контроль: два рази за семестр. Докладніша інформація щодо проведення та оцінювання наведена в PCO з дисципліни.

Семестровий контроль: залік. Загальна сума балів на заліку – 100 балів. Докладніша інформація щодо проведення та оцінювання наведена в PCO з дисципліни.

Умови допуску до семестрового контролю: семестровий рейтинг не нижче 50 балів, робота на практичних заняттях, виконання на позитивну оцінку експрес-тесту по МКР, тесту по ДКР та всіх тестів з лекційного матеріалу та практичних робіт. Якщо семестровий рейтинг є вищим за 60 балів за умови здачі всіх видів робіт студент може отримати залік автоматом.

Перевірка виконання CPC здійснюватиметься оцінюванням результатів тестування на платформі дистанційного навчання Google Workspace for Education Fundamentals на домені @LLL.kpi.ua в системі Google Клас за посиланням:

<https://classroom.google.com/c/NDA4MTY0OTYzOTA3?cjc=3ku5mbm>.

Кожен з тестів можливо пройти тільки один раз. Тести включають по 10 запитань різної складності, бали за кожне запитання залежать від його складності. Рейтингові (вагові) бали занять і рейтингові оцінки по всіх видах контролю в Google Класі дорівнюють відповідним балам в силабусі з коефіцієнтом 10 для зручності розрахунку балів (щоб не використовувати дробові числа).

Відповідно перед кожною атестацією, а також в кінці семестру всі набрані студентом бали в Google Класі будуть ділитися на 10 і вноситись до системи АІС «Електронний кампус» КПІ імені Ігоря Сікорського.

Студенти, які набрали вище 60 балів можуть отримати оцінку «автоматом». Студенти, які хочуть підвищити оцінку здають залік. Попередня оцінка анулюється.

7. Додаткова інформація з дисципліни (освітнього компонента)

Додаток 1

Система рейтингових (вагових) балів занять і рейтингових оцінок по видах контролю за рік

№ п/п	Вид контролю	Бал	Кількість	Сума балів
1.	Робота на практичних заняттях	3	15	45
2.	Тестування знань студентів з лекційного матеріалу та практичних робіт	5	7	35
3.	Модульна контрольна робота			
	-ваговий бал r_k	10	1	10
4.	ДКР			
	-ваговий бал r_k	10	1	10
5.				100

Розрахунок шкали (R) рейтингу

Сума вагових балів контрольних заходів протягом семестру складає:

$$R = 45 + 35 + 10 + 10 = 100 \text{ балів.}$$

Рейтингова шкала з дисципліни складає $R = 100$ балів.

Необхідною умовою для одержання заліку автоматом є робота на практичних заняттях, виконання на позитивну оцінку експрес-тесту по МКР, тесту по ДКР та всіх тестів з лекційного матеріалу та практичних робіт та загальний рейтинг не менше 60 балів. Для підвищення оцінки проводиться залікова робота. При цьому рейтинг отриманий за семестр анулюється.

Календарний контроль: два рази за семестр.

Рубіжні (планові атестації). Студент повинен набрати балів: 1 атестація – «зараховано» - 20 балів (40 – максимум), 2 атестація – 40 балів (80 – максимум).

Підсумкова оцінка якості знань з дисципліни визначаються за традиційною 6-рівневою шкалою на базі індивідуальних поточних оцінок за такою шкалою:

Кількість балів	Оцінка
100-95	Відмінно
94-85	Дуже добре
84-75	Добре
74-65	Задовільно
64-60	Достатньо
Менше 60	Незадовільно

Семестровий контроль: залік. Загальна сума балів заліку – 100 балів. Умови допуску до семестрового контролю: семестровий рейтинг не менше 50 балів, написання МКР та ДКР, виконання практичних робіт.

Залік складається з 10 питань, 1 питання оцінюється у 10 балів.
бездоганна робота

– 9-10 балів

є певні недоліки у виконанні роботи
є суттєві недоліки у виконанні роботи
відповіді не вірні

– 7-8 балів
– 2-6 балів
– 0-1 бал.

Таблиця відповідності рейтингових балів оцінкам за університетською шкалою:

<i>Кількість балів</i>	<i>Оцінка</i>
100-95	Відмінно
94-85	Дуже добре
84-75	Добре
74-65	Задовільно
64-60	Достатньо
Менше 60	Незадовільно

Додаток 2

Орієнтовний перелік питань для ДКР з курсу «Інформаційні технології»

1. Дайте визначення типу даних рядок в python та назвіть методи створення рядків.
2. Що таке зріз рядка в python?
3. Які форми зрізів Ви знаєте?
4. Яка функція python подібна до зрізу з трьома параметрами ?
5. Що таке сирі рядки в python, для чого вони використовуються?
6. Які Ви знаєте методи для роботи з рядками?
7. Дайте визначення типу даних списки в Python.
8. Які типи даних можуть включати списки в Python?
9. Назвіть способи створення списків в Python.
10. Як отримати доступ до елементів списку?
11. Які Ви знаєте методи для роботи зі списками?
12. Дайте визначення типу даних кортеж в python та назвіть методи створення типу даних кортеж.
13. Назвіть переваги кортежів.
14. Які Ви знаєте методи для роботи з кортежами?
15. Дайте визначення типу даних словник в python.
16. Назвіть методи створення типу даних словник в python.
17. Які Ви знаєте методи для роботи зі словниками?
18. Дайте визначення типу даних набори (множини) в python.
19. Які Ви знаєте особливості наборів (множин):
20. Для чого використовуються набори?
21. Які Ви знаєте методи створення множини в Python?
22. Які Ви знаєте методи для роботи з множинами?
23. Робота з масивами пакету numpy
24. Аналіз експериментальних даних з застосуванням пакету numpy:
 - середнє значення величини;
 - середню квадратичну похибку середнього;
 - середнє квадратичне відхилення величини;
 - дисперсію величини;
 - максимальне та мінімальне значення величини тощо.
25. Побудова графіків і діаграм на основі експериментальних даних.

Перелік питань до модульної контрольної роботи з курсу «Інформаційні технології»

1. Історія розвитку та основні мови програмування.
2. Об'єктно-орієнтований підхід до програмування.
3. Процедурний підхід до програмування.
4. Основні переваги Python. Задачі, для яких застосовують Python.
5. Типи даних в Python.
6. Логічні операції в Python.
7. Математичні оператори та функції в Python.
8. Функції в Python.
9. Класи в Python.
10. Основні пакети Python.
11. Основні функції роботи зі списками.
12. Основні функції роботи зі словниками.
13. Основні функції роботи з кортежами.
14. Основні функції роботи з множинами (наборами).
15. Визначення поняття біоінформатики.
16. Предмет біоінформатики.
17. Цілі біоінформатики.
18. Мета програми «Геном людини».
19. Коли стартувала програми «Геном людини» та основні її досягнення?
20. Коли використовується алгоритм локального вирівнювання для порівняння двох послідовностей?
21. Коли використовується алгоритм глобального вирівнювання для порівняння двох послідовностей?
22. Коли використовується алгоритм псевдоглобального вирівнювання для порівняння двох послідовностей?
23. Коли використовується алгоритм швидкого вирівнювання для порівняння двох послідовностей?
24. Що означає кожна комірка матриці ваг префіксів $a(i,j)$?
25. Основна ідея алгоритмів динамічного програмування?
26. Бази даних (загальне визначення) і біоінформаційні БД. В яких областях молекулярної біології застосовують БД.
27. Історія виникнення БД. Найбільш відомі організації - куратори БД і створені ними банки даних.
28. Класифікація БД. Наведіть приклади.
29. Які методи використовуються для вирівнювання 2-х послідовностей, коли і ким ці методи були розроблені?
30. Коли використовується алгоритм загальної функції штрафу для порівняння двох послідовностей?
31. Алгоритм загальної функції штрафу.
32. Ініціалізація масиву ваг префіксів $a(i,j)$; $b(i,j)$; $c(i,j)$ при використанні загальної функції штрафу.
33. Рекурентна формула для глобального вирівнювання двох послідовностей.
34. Рекурентна формула для локального вирівнювання двох послідовностей.
35. Рекурентна формула для псевдоглобального вирівнювання двох послідовностей.
36. Рекурентна формула для загальної функції штрафу.
37. Ініціалізувати нульовий стовпчик та нульову строку для глобального вирівнювання двох послідовностей.

38. Ініціалізувати нульовий стовпчик та нульову строку для локального вирівнювання двох послідовностей.
39. Ініціалізувати нульовий стовпчик та нульову строку для псевдоглобального вирівнювання двох послідовностей.
40. Ініціалізувати нульовий стовпчик та нульову строку для загальної функції штрафу.
41. Алгоритм хешування, в яких програмах він використовується?
42. Сформувати хеш-таблицю для дослідження послідовності (довжина слова 4).
43. Основні функції пакету Matplotlib для побудови графіків.
44. Основні функції пакету Opensv-python для роботи з зображеннями біологічних об'єктів графіків.
45. Основні засоби роботи з генетичними послідовностями в пакету Biopython.
46. Функції для лінійної регресії в пакеті Sklearn.

Додаток 4

Перелік питань до заліку з курсу «Інформаційні технології»

1. Історія розвитку та основні мови програмування.
2. Об'єктно-орієнтований підхід до програмування.
3. Процедурний підхід до програмування.
4. Основні переваги Python. Задачі, для яких застосовують Python.
5. Типи даних в Python.
6. Логічні операції в Python.
7. Математичні оператори та функції в Python.
8. Функції в Python.
9. Класи в Python.
10. Основні пакети Python.
11. Основні функції роботи зі списками.
12. Основні функції роботи зі словниками.
13. Основні функції роботи з кортежами.
14. Основні функції роботи з множинами (наборами).
15. Визначення поняття біоінформатики.
16. Предмет біоінформатики.
17. Цілі біоінформатики.
18. Мета програми «Геном людини».
19. Коли стартувала програми «Геном людини» та основні її досягнення?
20. Коли використовується алгоритм локального вирівнювання для порівняння двох послідовностей?
21. Коли використовується алгоритм глобального вирівнювання для порівняння двох послідовностей?
22. Коли використовується алгоритм псевдоглобального вирівнювання для порівняння двох послідовностей?
23. Коли використовується алгоритм швидкого вирівнювання для порівняння двох послідовностей?
24. Що означає кожна комірка матриці ваг префіксів $a(i,j)$?
25. Основна ідея алгоритмів динамічного програмування?
26. Бази даних (загальне визначення) і біоінформаційні БД. В яких областях молекулярної біології застосовують БД.
27. Історія виникнення БД. Найбільш відомі організації - куратори БД і створені ними банки даних.
28. Класифікація БД. Наведіть приклади.
29. Які методи використовуються для вирівнювання 2-х послідовностей, коли і ким ці методи були розроблені?

30. Коли використовується алгоритм загальної функції штрафу для порівняння двох послідовностей?
31. Алгоритм загальної функції штрафу.
32. Ініціалізація масиву ваг префіксів $a(i,j)$; $b(i,j)$; $c(i,j)$ при використанні загальної функції штрафу.
33. Рекурентна формула для глобального вирівнювання двох послідовностей.
34. Рекурентна формула для локального вирівнювання двох послідовностей.
35. Рекурентна формула для псевдоглобального вирівнювання двох послідовностей.
36. Рекурентна формула для загальної функції штрафу.
37. Ініціалізувати нульовий стовпчик та нульову строку для глобального вирівнювання двох послідовностей.
38. Ініціалізувати нульовий стовпчик та нульову строку для локального вирівнювання двох послідовностей.
39. Ініціалізувати нульовий стовпчик та нульову строку для псевдоглобального вирівнювання двох послідовностей.
40. Ініціалізувати нульовий стовпчик та нульову строку для загальної функції штрафу.
41. Алгоритм хешування, в яких програмах він використовується?
42. Сформувати хеш-таблицю для дослідження послідовності (довжина слова 4).
43. Основні функції пакету Matplotlib для побудови графіків.
44. Основні функції пакету Opencv-python для роботи з зображеннями біологічних об'єктів графіків.
45. Основні засоби роботи з генетичними послідовностями в пакеті Biopython.
46. Функції для лінійної регресії в пакеті Sklearn.

Робочу програму навчальної дисципліни (силабус):

складено д.т.н., проф. Горобець С.В.

Ухвалено кафедрою біоенергетики, біоінформатики та екобіотехнології (протокол № 19 від 27.06.2025 р.)

Погоджено Методичною комісією факультету (протокол № 6 від 30.06.2025).

Оновлено та погоджено Методичною комісією ФБТ (протокол №1 від 10.10.2025 р.) в частині умов отримання рейтингової семестрової оцінки.