



## 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	Заочна
Рік підготовки, семестр	2 курс, осінній семестр
Обсяг дисципліни	4 кредити (120 год.), в т.ч. лекцій – 4 год., практичних – 4 години, лабораторних – 0 годин, СРС – 112 год.
Семестровий контроль/ контрольні заходи	Залік/ МКР/ ДКР
Розклад занять	На сайтах <a href="http://roz.kpi.ua">http://roz.kpi.ua</a> та <a href="https://schedule.kpi.ua">https://schedule.kpi.ua</a>
Мова викладання	Українська
Інформація про керівника курсу / викладачів	Лектор: д. т. н., проф. Горобець Світлана Василівна, <a href="mailto:gorobetssv@gmail.com">gorobetssv@gmail.com</a> , телеграм чат дисципліни Практичні д. т. н., проф. Горобець Світлана Василівна, <a href="mailto:gorobetssv@gmail.com">gorobetssv@gmail.com</a> ,
Розміщення курсу	На платформі дистанційного навчання Google Клас за посиланням <a href="https://classroom.google.com/c/NDIwOTYxNTg1ODcy?cjc=qg4pebc">https://classroom.google.com/c/NDIwOTYxNTg1ODcy?cjc=qg4pebc</a> , код курсу qg4pebc, постійне посилання на Google Meet <a href="https://meet.google.com/yv-uzxi-bap">https://meet.google.com/yv-uzxi-bap</a> .

#### 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, особливо Biopython, що є колекцією некомерційних інструментів для обчислення в біології та біоінформатиці з відкритим кодом створеною міжнародною асоціацією розробників Biopython - 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. Даний курс знайомить майбутніх біотехнологів з основами біоінформатики та програмування на мові Python.

Метою навчальної дисципліни є формування у студентів здатностей:

- ФК-01: використовувати знання з математики та фізики в обсязі, необхідному для досягнення інших результатів освітньої програми;
- ФК-14: використовувати сучасні автоматизовані системи управління виробництвом біотехнологічних продуктів різного призначення, їх технічне, алгоритмічне, інформаційне і програмне забезпечення для вирішення професійних завдань;
- ФК-16: здатність комплексно аналізувати біологічні та біотехнологічні процеси на молекулярному та клітинному рівнях.

Після засвоєння навчальної дисципліни студенти мають продемонструвати такі результати навчання:

Програмні результати навчання:

- ПРН-01: Вміти застосовувати сучасні математичні методи для розв'язання практичних задач, пов'язаних з дослідженням і проектуванням біотехнологічних процесів. Використовувати знання фізики для аналізу біотехнологічних процесів.
- ПРН-15: Базуючись на знаннях про закономірності механічних, гідромеханічних, тепло- та масообмінних процесів та основні конструкторські особливості, вміти обирати відповідне устаткування у процесі проектування виробництв біотехнологічних продуктів різного призначення для забезпечення їх максимальної ефективності.

## 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.*

Різні мови програмування та сфери їх застосування. Об'єктно-орієнтований і системний підходи. Порівняння об'єктно-орієнтованого та процедурного (традиційного) підходів. Особливості мови Python. Пакети Python для вирішення задач біотехнології та біоінформатики.

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

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

Робота з файлами в мові Python.

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

Пакет Matplotlib для побудови графіків та діаграм.

Пакет Sklearn для задач регресійного аналізу. Лінійна регресія експериментальних даних.

Пакет OpenCV-python для аналізу зображень, отриманих методами оптичної мікроскопії, скануючої зондової мікроскопії, скануючої електронної мікроскопії. Зчитування зображень біооб'єктів із файлів, їх запис у файл, використання фільтрів для корекції зображень, виділення контурів біооб'єктів тощо.

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

Формування біоінформатики як самостійної науки. Предмет і задачі біоінформатики.

Методи динамічного програмування. Алгоритм глобального вирівнювання двох послідовностей.

Функції, класи та проекти в Python та приклад їх реалізації в алгоритмі глобального вирівнювання двох послідовностей.

Основи пакету Biopython для роботи з базами даних.

Модифікації алгоритму вирівнювання генетичних послідовностей. Алгоритми локального та псевдоглобального вирівнювання двох нуклеотидних послідовностей.

Модифікації алгоритму вирівнювання генетичних послідовностей. Алгоритми швидкого вирівнювання двох нуклеотидних послідовностей. Алгоритм побудови точкової матриці (dot-матриці).

Система премій і штрафів для вирівнювання послідовностей. Алгоритм загальної функції штрафу. Хешування. Використання хеш-таблиці для пошуку співпадінь по списку слів.

Особливості алгоритмів вирівнювання білкових послідовностей. Матриці амінокислотних замінів PAM і BLOSUM.

Використання пакету Biopython для дослідження властивостей ДНК, РНК та білків на прикладі вірусу COVID-19.

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.
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. Замуруєва О. В., Кримусь А. С., Ольхова Н. В. Об'єктно-орієнтоване програмування в Python: курс лекцій. Луцьк : Вежа-Друк, 2018. – 64 с.
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.

Додаткова:

1. Костюченко А.О. Основи програмування мовою Python: навчальний посібник. Ч.: ФОП Баликіна С.М., 2020. 180 с.
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.

3. Bioinformatics. - editor: Nakaya Helder I., PhD, editor Information Brisbane (AU): 2021. – 188 p.
4. Lesk A.M. Introduction to Bioinformatics. – Oxford. – 2002. – 255 p.

### 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](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	<b>Лекція 1.</b> Різні мови програмування та сфери їх застосування. Об'єктно-орієнтований і системний підходи. Порівняння об'єктно-орієнтованого та процедурного (традиційного) підходів. Література: базова [1,5], додаткова [1,2], інформаційні джерела [2-6]
2	<b>Лекція 2.</b> Предмет і задачі біоінформатики. Методи динамічного програмування. Алгоритм глобального та локального вирівнювання двох послідовностей та приклад реалізації алгоритму глобального вирівнювання двох послідовностей на мові програмування Python. Література: базова [2-4], інформаційні джерела [1]

#### Практичні заняття

	2
1.	<b>МКР</b>
2.	<b>Залік</b>

#### Самостійна робота студента

Самостійна робота студента по дисципліні включає підготовку до аудиторних занять (6 годин), модульної контрольної (4 години), написання ДКР (10 годин), підготовка до заліку (6 годин), а також опрацювання наведених нижче тем самостійно (86 годин).

№	Назви тем для самостійного вивчення	Години
1	Типи даних в Python. Математичні операції, умовні оператори, цикли. Зчитування та запис файлів. Функції, класи та проєкти у Python. Робота з файлами в мові Python. Література: базова [1,5], додаткова [1,2], інформаційні джерела [2]	12
2	Пакет Matplotlib для побудови графіків та діаграм. Пакет Sklearn для задач регресійного аналізу. Лінійна регресія експериментальних даних. Література: базова [1,5], інформаційні джерела [3]	12
3	Пакет Opencv-python для аналізу зображень,	12

	отриманих методами оптичної мікроскопії, скануючої зондової мікроскопії, скануючої електронної мікроскопії. Зчитування зображень біооб'єктів із файлів, їх запис у файл, використання фільтрів для корекції зображень, виділення контурів біооб'єктів тощо. Література: базова [1,5], інформаційні джерела [2,5]	
4	Основи пакету Віорутон для роботи з базами даних. Література: базова [1,5], інформаційні джерела [6]	6
5	Формування біоінформатики як самостійної науки. Предмет і задачі біоінформатики. Біоінформатика як інструмент отримання нових знань. Література: базова [2-4], інформаційні джерела [1]	2
6	Методи динамічного програмування. Алгоритм швидкого, алгоритм загальної функції штрафу вирівнювання двох послідовностей. Особливості алгоритмів вирівнювання білкових послідовностей. Матриці амінокислотних замінів РАМ і ВЛОСУМ. Хешування. Використання хеш-таблиці для пошуку співпадіннь по списку слів. Література: базова [2-4], інформаційні джерела [1]	30
7	Використання пакету Віорутон для дослідження властивостей ДНК, РНК та білків на прикладі вірусу COVID-19. Література: базова [1,5], інформаційні джерела [6]	12

## 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 6 силабуса.

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

- *правила відвідування занять (як лекцій, так і практичних/лабораторних);*  
Відвідування лекцій, практичних занять, 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 студент може отримати до 4 балів сумарно протягом семестру.
- *політика дедлайнів та перекладань;*  
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 онлайн тестування та виконання розрахунків.  
*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/NDIwOTYxNTg1ODcy?cjc=qg4pebc> . 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/](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 учням 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 учнів. 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 qg4pebc. 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:** відсутній.

**Календарний контроль:** проводиться в кінці семестру.

**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:** написання МКР та виконання ДКР. Якщо семестровий рейтинг є не менше 60 балів, за умови написання МКР та виконання ДКР, студент може отримати залік автоматом.

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

№ п/п	Вид контролю	Бал	Кількість	Сума балів
1.	Модульна контрольна робота			
	-ваговий бал $r_k$	50	1	50
	- якість виконання*	0-50		
2.	ДКР			
	-ваговий бал $r_k$	50	1	50
	- якість виконання**	0-50		
5.				100

\* - Якість виконання модульної контрольної роботи. Контрольна робота складається з 5 питань. Кожне питання оцінюється у 10 балів:

повна розкрита відповідь – 9-10 балів ;  
 помилка в одному завданні або неповна відповідь в двох завданнях – 8 балів ;  
 помилка в двох завдань або неповна відповідь в 4 завданнях – 7 балів;  
 робота не захищена – 0-6 балів.

\*\* - Якість виконання домашньої контрольної роботи. Домашня контрольна робота складається з 5 питань. Кожне питання оцінюється у 10 балів:

повна розкрита відповідь – 9-10 балів ;  
 помилка в одному завданні або неповна відповідь в двох завданнях – 8 балів ;  
 помилка в двох завдань або неповна відповідь в 4 завданнях – 7 балів;  
 робота не захищена – 0-6 балів.

**Розрахунок шкали (R) рейтингу**

Сума вагових балів контрольних заходів протягом семестру складає:

$$R = 50 + 50 = 100 \text{ балів};$$

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

Необхідною умовою для одержання заліку автоматом є зарахування усіх пропозицій, що виносяться на обговорення виконання на позитивну оцінку модульної контрольної роботи та загальний рейтинг більше 60 балів. Для підвищення оцінки проводиться залікова робота. При цьому попередній рейтинг анулюється.

**Календарний контроль: відсутній**

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

Рейтинг	Оцінка ESTS	Традиційна оцінка
$95 \leq R < 100$	A	Відмінно
$85 \leq R < 94$	B	Дуже добре
$75 \leq R < 84$	C	Добре
$65 \leq R < 74$	D	Задовільно
$60 \leq R < 64$	E	Достатньо
$R < 60$	Fx	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.

Заліковий білет складається з 5 питань, 1 питання оцінюється у 20 балів.

Повна відповідь на питання – (10) балів

Повна відповідь на питання – (20) балів  
Зроблені незначні помилки – (18-19) балів  
Суттєві помилки у відповіді – (16-17) балів  
Відповіді не вірні – (0-15) бали.

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 тем ДКР**

1. Вбудовані функції Python
2. Вбудовані константи Python
3. Вбудовані типи даних Python
4. Вбудовані виключення Python
5. Модулі Python
6. Модулі для обробки тексту Python
7. Модулі для роботи з різними типами даних Python
8. Числові та математичні модулі Python
9. Модулі для функціонального програмування Python
10. Модулі доступу до файлів і директорій Python
11. Збереження даних Python
12. Стиснення та архівація даних Python
13. Формати файлів Python
14. Модулі для взаємодії з операційною системою Python
15. Сервіси періоду виконання Python
16. Модулі для програмних фреймворків Python
17. Модулі для створення графічного інтерфейсу користувача з Tk Python
18. Криптографічні сервіси Python
19. Модулі для обробки даних Інтернету Python
20. Модулі обробки структурованої розмітки Python
21. Модулі для підтримки Інтернет-протоколів Python
22. Модулі для мультимедійних послуг Python
23. Модулі для інтернаціоналізації Python
24. Модулі для засобів розробки Python
25. Модулі для відладки та профілювання Python
26. Модулі для публікації та встановлення програмного забезпечення Python

Додаток 3

**Питання до 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 БІБД.
27. Історія виникнення БІБД. Найбільш відомі організації - куратори БІБД і створені ними банки даних.
28. Класифікація БІБД. 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 Orncsv-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

### Питання 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 пакету 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 № 14 від 27.05.2024 р.)

Погоджено Методичною комісією факультету (протокол №19 від 28.06.2024).