



## ОСНОВИ ІНФОРМАТИКИ ТА ПРОГРАМУВАННЯ (ПО 17)

### Робоча програма навчальної дисципліни (Силлабус)

#### 1. Реквізити навчальної дисципліни

Рівень вищої освіти	<i>Перший (бакалаврський)</i>
Галузь знань	<i>12 Інформаційні технології</i>
Спеціальність	<i>122 Комп'ютерні науки</i>
Освітня програма	<i>Комп'ютерні технології в біології та медицині</i>
Статус дисципліни	<i>Нормативна</i>
Форма навчання	<i>очна(денна)</i>
Рік підготовки, семестр	<i>1 курс, осінній семестр</i>
Обсяг дисципліни	<i>5 кредити ЄКТС/ 150 год. Лекційні заняття – 18 год., комп'ютерні практикуми – 54 год, домашня контрольна робота -10год, СР-78</i>
Семестровий контроль/ контрольні заходи	<i>Екзамен / модульна контрольна робота /Домашня контрольна робота</i>
Розклад занять	<i>Лекції (два рази кожного тижня починаючи з 1-го по 15 тиждень) Комп'ютерні практикуми (два рази кожного тижні бажано після лекції) Для заочників не передбачено</i>
Мова викладання	<i>Українська</i>
Інформація про керівника курсу / викладачів	<i>Лектор: доцент, к.ф.-м.н, доцент каф. БМК, Федорін Ілля Валерійович, (095) 322-72-50; <a href="mailto:fedorin.illia@lil.kpi.ua">fedorin.illia@lil.kpi.ua</a> Комп'ютерні практикуми 1. доцент, к.т.н., доцент каф. БМК Кравченко Ольга Віталіївна (067) – 911-83-33 (телеграм) <a href="mailto:bmkravchenko-fbmi@lil.kpi.ua">bmkravchenko-fbmi@lil.kpi.ua</a></i>
Розміщення курсу	<i>wyupcsnz (Google classroom)</i>

#### 2. Програма навчальної дисципліни

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

Основою вивчення кредитного модуля є процес навчання і підготовки фахівця зі спеціальності 122 «Комп'ютерні науки» за ОПП «Комп'ютерні технології в біології та медицині» першого (бакалаврського) рівня вищої освіти ступеня бакалавра здатного вирішувати задачі з галузі комп'ютерних наук, пов'язаних з розробкою програмного продукту на основі мови Python для налаштування роботи баз даних в сфері біології і медицини.

**Предметом** вивчення навчальної дисципліни є вивчення можливостей використання мови Python при розробці програмного продукту для налаштування роботи баз даних та більш широких можливостей застосування пакету програмного забезпечення Microsoft Office в оформленні звітів, технічної документації, індивідуальних завдань тощо.

**Мета кредитного модуля** є формування у студентів знань та умінь використовувати методи та засоби програмування на основі мови Python згідно з новітніми принципами і тенденціями створення програмних продуктів налаштування продуктивності роботи баз даних та більш широко використовувати можливості інформаційного пакету програмного забезпечення Microsoft Office в подальшій навчальній та професійній діяльності студента.

Відповідно до освітньо-професійної програми (ОПП -2022) «Комп'ютерні технології в біології та медицині» першого «бакалаврського» рівня вищої освіти після вивчення дисципліни студенти мають набути наступних **компетентностей**.

### **Інтегральні компетенції.**

Здатність розв'язувати складні спеціалізовані задачі та практичні проблеми у галузі комп'ютерних наук або у процесі навчання, що передбачає застосування теорій та методів комп'ютерних наук, інформаційних технологій і характеризується комплексністю та невизначеністю умов.

### **Загальні компетенції**

**ЗК 1** Здатність до абстрактного мислення, аналізу та синтезу

**ЗК 11** Здатність приймати обґрунтовані рішення

### **Фахові компетенції**

**ФК 3** 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.

**ФК 8** Здатність проектувати та розробляти програмне забезпечення із застосуванням різних парадигм програмування: узагальненого, об'єктно-орієнтованого, функціонального, логічного, з відповідними моделями, методами й алгоритмами обчислень, структурами даних і механізмами управління.

**ФК 18** Здатність проектувати та реалізовувати програмне забезпечення для вирішення різноманітних задач в біології та медицині із застосуванням різних мов програмування та сучасних бібліотек крос-платформного програмування, проводити тестування на декількох апаратних платформах, впроваджувати і підтримувати роботу інформаційних систем на сучасних платформах

В результаті засвоєння кредитного модуля студенти мають продемонструвати такі **програмні результати навчання**:

**ПРН 9** 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

**ПРН 13** Володіти мовами системного програмування та методами розробки програм, що взаємодіють з компонентами комп'ютерних систем

## **2. Пререквізити та пост реквізити дисципліни (місце в структурно-логічній схемі навчання за відповідною освітньою програмою)**

В структурно-логічній схемі програми підготовки фахівця навчальна дисципліна «Основи інформатики та програмування» входить до переліку нормативних дисциплін, спрямованих на формування спеціальних компетенцій фахівця.

Пререквізити – навчальна дисципліна викладається в 1-му семестрі 1-го року навчання та не залежить від інших навчальних дисциплін у структурно-логічній схемі освітньої програми.

Пост Реквізити - дана навчальна дисципліна формує навички у студентів із освітньо-професійної програми «Комп'ютерні технології в біології та медицині» першого (бакалаврського) рівня вищої освіти та є основою для забезпечення наступних навчальних дисциплін (кредитних модулів): Алгоритмізація та програмування, Об'єктно-орієнтоване програмування.

Кредитний модуль є основою для підготовки дипломних робіт за ОПП та в подальшій практичній роботі за фахом

## **3. Зміст навчальної дисципліни**

### **Розділ 1. Базові поняття програмування на мові Python**

#### **Тема 1.1. Історія розвитку обчислювальної техніки**

Тема 1.2. <i>Поняття алгоритму. Алгоритмічні структури</i>
Тема 1.3 <i>Вступ до програмування на Python. Комп'ютерна арифметика та робота на Python в консольному режимі</i>
Тема 1.4 <i>Типи даних</i>
Тема 1.5 <i>Алгоритмічні структури в мові Python. Керування логічною послідовністю виконання програми. Цикли while</i>
Тема 1.6 <i>Складні структури даних: рядок, кортеж, список, словник. Оператор for.</i>
Тема 1.7 <i>Процедури, функції та модульність в Python</i>
Тема 1.8 <i>Файли. Обробка винятків.</i>
<b>Розділ 2. Основи інформатики</b>
<i>Основи інформатики</i>

#### 4. Навчальні матеріали та ресурси

Для підготовки до лекційних, комп'ютерних практикумів, модульної контрольної роботи, індивідуального завдання самостійної роботи тощо використовується базова та додаткова література (надалі – література), а також рекомендовані методичні рекомендації. Література, яку треба використовувати для опанування дисципліни, опрацьовується студентами самостійно із застосуванням інтернет-ресурсів, Google classroom. В умовах дистанційного навчання можна готуватись за допомогою розміщеної в е-вигляді літературою в Google classroom з навчальної дисципліни.

##### 4.1 Базовою літературою

1. Основи інформатики та програмування. Конспект лекцій [Електронний ресурс] : навчальний посібник для здобувачів першого (бакалаврського) рівня вищої освіти за освітньою програмою «Комп'ютерні технології в біології та медицині» спеціальності 122 «Комп'ютерні науки» / КПІ ім. Ігоря Сікорського ; уклад.: І. В. Федорін. – Електронні текстові дані (1 файл: 1,09 Мбайт). – Київ : КПІ ім. Ігоря Сікорського, 2023. – 179 с. Режим доступу: <https://ela.kpi.ua/handle/123456789/63428>
2. Основи інформатики та програмування. Практикум [Електронний ресурс] : навчальний посібник для здобувачів першого (бакалаврського) рівня вищої освіти за освітньою програмою «Комп'ютерні технології в біології та медицині» спеціальності 122 «Комп'ютерні науки» / КПІ ім. Ігоря Сікорського ; уклад.: І. В. Федорін. – Електронні текстові дані (1 файл: 107 Кбайт). – Київ : КПІ ім. Ігоря Сікорського, 2023. – 89 с. Режим доступу: <https://ela.kpi.ua/handle/123456789/63429>
3. Федорін, І. В. Проектування та аналіз обчислювальних алгоритмів: вступ до алгоритмів [Електронний ресурс] : навчальний посібник для здобувачів першого (бакалаврського) рівня вищої освіти за освітньою програмою «Комп'ютерні технології в біології та медицині» спеціальності 122 «Комп'ютерні науки» / І. В. Федорін ; КПІ ім. Ігоря Сікорського. – Електронні текстові дані (1 файл: 1,98 Мбайт). – Київ : КПІ ім. Ігоря Сікорського, 2022. – 116 с. Режим доступу: <https://ela.kpi.ua/handle/123456789/50935>
4. Федорін, І. В. Проектування та аналіз обчислювальних алгоритмів: структури даних [Електронний ресурс] : навчальний посібник для для здобувачів першого (бакалаврського) рівня вищої освіти за освітньою програмою «Комп'ютерні технології в біології та медицині» спеціальності 122 «Комп'ютерні науки» / І. В. Федорін ; КПІ ім. Ігоря Сікорського. – Електронні текстові дані (1 файл: 2,8 Мбайт). – Київ : КПІ ім. Ігоря Сікорського, 2022. – 174 с. Режим доступу: <https://ela.kpi.ua/handle/123456789/50937>
5. Костюченко А. О. Основи програмування мовою Python: навчальний посібник. Ч.: ФОП Баликіна С.М., 2020. 180 с.

Режим

доступу:

<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](#)

#### 4.2. Додаткова література

1. Ерік М. Пришвидшений курс Python / Маттес Ерік. – Київ: Видавництво Старого Лева, 2021. – 600 с.
2. Васильєв О. Програмування мовою Python / Олексій Васильєв. – Тернопіль: Навчальна книга Богдан, 2019. – 204 с.
3. Naomi Ceder The Quick Python Book 3rd Edition / Naomi Ceder. – NY: Manning Publications Co., 2018 – 432 p.
4. Kenneth A. Lambert Fundamentals of Python: first programs / Kenneth A. Lambert. – NY: Cengage Learning, 2018 – 476 p.
5. Кренивич А. П. Python у прикладах і задачах. Частина 1. Структурне програмування Навчальний посібник із дисципліни "Інформатика та програмування" – К.: ВПЦ "Київський Університет", 2017. – 206 с.
6. Mark Lutz, Learning Python, 4<sup>th</sup> Edition. – Sebastopol, CA: O'Reilly Media, Inc., 2009. – 1161 p.
7. Дунець Р. Б. Арифметичні основи комп'ютерної техніки / Р. Б. Дунець, О. Т. Кудрявцев. – Львів : Ліга-Прес, 2006. – 142 с.
8. Довгалець С. М. Алгоритмічні мови та програмування. Частина 1. Основи інформатики та комп'ютерної техніки. Навчальний посібник / С. М. Довгалець, Р. В. Маслій. – Вінниця : ВНТУ, 2009. – 116 с.
9. Яковенко А. Основи програмування: методичні вказівки до виконання комп'ютерних практикумів з дисципліни "Основи програмування". Основи програмування мовою Python / А. В. Яковенко. – Київ : НТУУ "КПІ ім. І. Сікорського", 2017. – 87 с.
10. Електронний путівник мовою програмування Python. [Режим доступу]: <https://pythonguide.rozh2sch.org.ua/>
11. Онлайн платформа та БД для вивчення алгоритмів та мов програмування. [Режим доступу] <https://leetcode.com/>
12. Алгебра логіки. Матеріал з вікіпедії [Режим доступу]: [https://uk.wikipedia.org/wiki/Алгебра\\_логіки](https://uk.wikipedia.org/wiki/Алгебра_логіки)
13. Досконалі диз'юнктивна й кон'юнктивна нормальні форми. Лабораторна робота. [Режим доступу]: <http://www.berkut.mk.ua/download/knyrik/dm/lab1.pdf>
14. The Python Tutorial [Електронний ресурс] – Режим доступу до ресурсу: <https://docs.python.org/3/tutorial/index.html>.
15. Навчальні матеріали: Python [Електронний ресурс] – Режим доступу до ресурсу: <http://www.matfiz.univ.kiev.ua/pages/13>.
16. Кренивич, А.П. С у задачах і прикладах : навчальний посібник із дисципліни "Інформатика та програмування" / А.П. Кренивич, О.В. Обвінцев. – К. : Видавничо-поліграфічний центр "Київський університет", 2011. – 208 с.
17. Збірник задач з дисципліни "Інформатика і програмування" / Вакал Є.С., Личман В.В., Обвінцев О.В., Бублик В.В., Довгий Б.П., Попов В.В. -2- ге видання, виправлене та доповнене –К.: ВПЦ "Київський університет", 2006.– 94 с.
18. E-Olymp [Електронний ресурс] – Режим доступу до ресурсу: [www.eolymp.com](http://www.eolymp.com).
19. Eric Matthes, Python Crash Course, 2nd Edition: A Hands-On, Project-Based Introduction to Programming, 2019, 544 p.

20. Paul Barry, Head First Python: A Brain-Friendly Guide Paperback, O'Reilly Media,- 2016 - 2nd Edition, 622p.
21. Zed A. Shaw. 2017. Learn Python 3 the Hard Way: A Very Simple Introduction to the Terrifyingly Beautiful World of Computers and Code (1st. ed.). Addison-Wesley Professional.
22. Narasimha Karumanchi, Data Structure and Algorithmic Thinking with Python: Data Structure and Algorithmic Puzzles, 2015, 476p.
23. Brian Draper, Python Programming: A Complete Guide for Beginners to Master, Python Programming Language: A Complete Guide For Beginners To Master And Become An Expert In Python Programming Language, 2016, 127p.
24. Al Sweigart, Automate the Boring Stuff with Python, 2nd Edition: Practical Programming for Total Beginners, No Starch Press, Inc., 2020, 592p.

#### 4.3. Інформаційні ресурси

25. Методичні рекомендації до виконання комп'ютерних практикумів (поточна версія: <http://ela.kpi.ua/handle/123456789/19848> ).
26. Електронний портал інформатики та програмування – <https://www.geeksforgeeks.org/>
27. Електронний путівник мовою програмування Python – <https://pythonguide.rozh2sch.org.ua/>
28. Автоматизована інформаційна система «Електронний кампус НТУУ «КПІ» - <http://kpi.ua/ecampus>
29. Навчальна БД: <http://witdba.iptcom.net:8080/apex/>
30. Онлайн платформа та БД для вивчення алгоритмів та мов програмування – <https://leetcode.com/>

### 3. Навчальний контент

#### 5. Методика опанування навчальної дисципліни (освітнього компонента)

Під час вивчення кредитного модуля заплановано проведення 9 лекційних та 27 комп'ютерних практикумів (виконання загалом 11 практикумів), виконання модульної контрольної роботи та індивідуального завдання.

Під час вивчення навчального матеріалу застосовуються наступні **методи навчання**:

На **лекційних заняттях** педагог у словесній формі розкриває сутність наукових понять, явищ, процесів, логічно пов'язаних, об'єднаних загальною темою.

Ефективність проведення навчання на лекціях неможливе без широкого використання наочних методів. Вони зумовлені діалектичними закономірностями пізнання і психологічними особливостями сприймання. Наочний метод передбачає використання на лекціях з кредитного модуля *пояснювально-ілюстративний метод*.

При цьому студенти мають розуміти, що основне джерело отримання наукової інформації — не викладач, а книга. Тому важливо щоб студенти самостійно роботи з книгою: читали, конспектували додаткову інформацію до лекційного матеріалу.

Ведення конспекту дає змогу студенту:

- краще підготуватись до екзамену з кредитного модуля;
- вирішити спірних питань з відповіді (не повну, неточну відповідь) студента на екзамені;

На **комп'ютерних практикумах** важливе місце займає *інструктаж*. Він передбачає розкриття норм поведінки, особливостей використання методів і навчальних засобів, дотримання правил безпеки під час виконання навчальних операцій. При цьому важливо, щоб студенти розуміли не лише, *що* треба робити, а і як це робити.

Ефективність проведення **комп'ютерних практикумів з Розділу 1** багато в чому зумовлено способом організації мислення студентів. При цьому застосовується *Частково-пошуковий* метод проведення занять.

Застосований *Частково-пошуковий* метод проведення занять сприяє до активного пошуку розв'язання поставлених завдань, над якими студенти працюють самостійно під керівництвом педагога або на основі методичних вказівок або презентацій. Процес мислення студента при цьому набуває продуктивного характеру, але при цьому поетапно спрямовується і контролюється педагогом або самими студентами на основі роботи над виконанням комп'ютерного практикуму.

Ефективність проведення **комп'ютерних практикумів з Розділу 2** в основному полягає в застосуванні *практичного та репродуктивного методу* проведення занять. При практичному методі студенти отримують знання й уміння, виконувати практичні дії з виконання комп'ютерних практикумів з розділу 2 та оформлення індивідуального завдання. Репродуктивний метод на основі методичних рекомендацій та прикладів виконання завдань надають студентам можливість сформувати знання, навички та вміння з виконання поставлених завдань в розділі 2.

Нижче наведено розподіл аудиторних годин за темами курсу та календарний план їх проведення

Назви розділів і тем	Лекції		Семінарські		Оцінюванн я	Програмні результати навчання
	Год-ни	Тиж-ні	Год-ни	Тиж-ні		
<b>Розділ 1. Базові поняття програмування на мові Python</b>						
<i>Тема 1.1. Історія розвитку обчислювальної техніки</i>	2	1	4	3,4	КП№1	ПРН-9
<i>Тема 1.2. Поняття алгоритму. Алгоритмічні структури</i>	2	3	4	4-5	КП№2	ПРН-9
<i>Тема 1.3 Вступ до програмування на Python. Комп'ютерна арифметика та робота на Python в консольному режимі</i>	2	5	4	6-7	КП№3	ПРН-9
<i>Тема 1.4 Типи даних</i>	2	7	6	7-8		ПРН-9, 13
<i>Тема 1.5 Алгоритмічні структури в мові Python. Керування логічною послідовністю виконання програми. Цикли while</i>	2	9	6	9, 10	КП№4 КП№5	ПРН-9, 13
<i>Тема 1.6 Складні структури даних: рядок, кортеж, список, словник. Оператор for.</i>	2	11	6	11, 12	КП№6 КП№7 КП№8 КП№9	ПРН-9, 13
<i>Тема 1.7 Процедури, функції та модульність в Python</i>	4	13-15	6	13, 14	КП№10	ПРН-9, 13
<i>Тема 1.8 Файли. Обробка винятків.</i>	2	17	6	15, 16	КП№11	ПРН-9, 13
<i>Модульна контрольна робота</i>			2	17	МКР	ПРН-9, 13
<b>Разом за розділом 1</b>	<b>18</b>		<b>44</b>			
<b>Розділ 2. Основи інформатики</b>						
<b>Основи інформатики</b>			10	3, 9, 11, 13,15	КП№12 КП№13 КП№14 КП№15	ПРН-9
<b>Разом за розділом 2</b>			<b>10</b>			
<i>ДКР</i>				18	ДКР	ПРН-9, 13
<i>Екзамен</i>	<i>За графіком</i>					ПРН-9, 13
<b>Всього годин</b>	<b>18</b>		<b>54</b>			

Відповідність методів навчання та оцінювання відображені в рейтинговій системі оцінювання, яка передбачає: захист звітів з комп'ютерних практикумів (КП), модульну контрольну роботу, індивідуальне завдання та підсумковий екзамен.

### 5.1.Лекційні заняття

№ з/п	Назва теми лекції та перелік основних питань (перелік дидактичних засобів, завдання на СРС з посиланням на літературу)
1	<p><b>I. Історія розвитку обчислювальної техніки</b>  <i>Історія обчислювальної техніки. Двійкова система числення. Будова персонального комп'ютера. Мови програмування Python</i>  <b>Завдання на СРС</b>            1. Вивчення лекційного матеріалу. Вивчення рекомендованої літератури. Вивчення термінів і основних понять з теми. Підготовка до комп'ютерного практикуму і розробка звіту з лабораторної роботи [1 - 3, 5 - 7]</p> <p><b>II. Основи алгоритмів</b>  <i>Поняття алгоритму. Властивості алгоритму. Способи представлення алгоритмів. Основні структури алгоритмів</i>  <b>Завдання на СРС</b>            1. Вивчення лекційного матеріалу. Вивчення рекомендованої літератури. Вивчення термінів і основних понять з теми. Підготовка до комп'ютерного практикуму і розробка звіту з лабораторної роботи [1 - 3, 5 - 7]</p>
2	<p><b>Базові поняття програмування в системі Python</b>  <i>Особливості програмування в системі Python. Версії мови Python. Встановлення системи Python на комп'ютері. Діалоговий режим роботи Python</i>  <b>Завдання на СРС</b>            1. Вивчення лекційного матеріалу. Вивчення рекомендованої літератури. Вивчення термінів і основних понять з теми. Підготовка до комп'ютерного практикуму і розробка звіту з лабораторної роботи [1 - 3, 5 - 7]            2. <b>Тема 1.2.</b> Поняття алгоритму. Алгоритмічні структури  <i>Поняття алгоритму. Властивості алгоритму. Способи представлення алгоритмів. Основні структури алгоритмів</i>  <b>Література</b>            основна [1 - 3]; додаткова [4]</p>
3	<p><b>Комп'ютерна арифметика та робота на Python в діалоговому режимі</b>  <i>Загальні поняття комп'ютерної арифметики. Цілі числа та числа з "плаваючою крапкою". Консольна робота з арифметичними виразами в Python. Поняття змінної та операції присвоювання в Python. Значення None в Python</i>  <b>Завдання на СРС</b>            1. Вивчення лекційного матеріалу. Вивчення рекомендованої літератури. Вивчення термінів і основних понять з теми. Підготовка до комп'ютерного практикуму і розробка звіту з лабораторної роботи [1 - 3, 5 - 7]</p>
4	<p><b>Типи даних</b>            1) Ініціалізація змінних. Арифметичні операції в Python. Перевірка типу даних змінної в Python.            2) Логічний тип даних. Логічні вирази  <b>Завдання на СРС</b>            1. Вивчення лекційного матеріалу. Вивчення рекомендованої літератури. Вивчення термінів і основних понять з теми. Підготовка до комп'ютерного практикуму і розробка звіту з лабораторної роботи [1 - 3, 5 - 7]</p>
5	<p><b>Керування логічною послідовністю виконання програми</b>  <i>Керування логічною послідовністю виконання програми. Тип "логічний" та операції над даними цього типу. Таблиці істинності базових логічних операцій. Логічні операції в керуванні процесу виконання програми, оператори if-elif-else. Початкові поняття циклів роботи програми за допомогою оператора while</i></p>

№ з/п	Назва теми лекції та перелік основних питань (перелік дидактичних засобів, завдання на СРС з посиланням на літературу)
	<p><b><u>Завдання на СРС</u></b> 1. Вивчення лекційного матеріалу. Вивчення рекомендованої літератури. Вивчення термінів і основних понять з теми. Підготовка до комп'ютерного практикуму і розробка звіту з лабораторної роботи [1 - 3, 5 - 7]</p>
6	<p><b><i>Організація циклів за допомогою оператора while</i></b> <i>Види циклів: перевірка закінчення циклу в кінці циклу, на початку циклу, безпосередньо в умовній частині оператора while. Прямі та зворотні відношення порядку та особливості їх застосування при організації циклу.</i></p> <p><b><u>Завдання на СРС</u></b> 1. Вивчення лекційного матеріалу. Вивчення рекомендованої літератури. Вивчення термінів і основних понять з теми. Підготовка до комп'ютерного практикуму і розробка звіту з лабораторної роботи [1 - 3, 5 - 7]</p>
7	<p><b><i>I. Складні структури даних: рядок, кортеж, список, словник</i></b> <i>Базові поняття складних структур даних (послідовності): список. Незмінні та змінні типи даних в Python. Базові методи доступу до складових складних типу даних список. Базові операції зі списками. Основні функції роботи зі списками та функції перетворення типів.</i></p> <p><b><u>Завдання на СРС</u></b> 1. Вивчення лекційного матеріалу. Вивчення рекомендованої літератури. Вивчення термінів і основних понять з теми. Підготовка до комп'ютерного практикуму і розробка звіту з лабораторної роботи [1 - 3, 5 - 7]</p> <p><b><i>II. Складні структури даних: рядок, кортеж, список, словник</i></b> <i>Базові поняття складних структур даних (послідовності): рядок. Базові методи доступу до складових типу даних рядок. Базові операції із рядками. Основні функції роботи з рядками та функції перетворення типів.</i></p> <p><b><u>Завдання на СРС</u></b> 1. Вивчення лекційного матеріалу. Вивчення рекомендованої літератури. Вивчення термінів і основних понять з теми. Підготовка до комп'ютерного практикуму і розробка звіту з лабораторної роботи [1 - 3, 5 - 7]</p>
8	<p><b><i>Складні структури даних: рядок, кортеж, список, словник</i></b> <i>Тип даних "словник". Використання словників та списків для побудови баз даних</i></p> <p><b><u>Завдання на СРС</u></b> 1. Вивчення лекційного матеріалу. Вивчення рекомендованої літератури. Вивчення термінів і основних понять з теми. Підготовка до комп'ютерного практикуму і розробка звіту з лабораторної роботи [1 - 3, 5 - 7]</p>
9	<p><b><i>Оператор for в Python</i></b> <i>Побудова циклів в Python за допомогою оператора for. Генератори списків та генерація списків за допомогою оператора for. Особливості використання змінної циклу при роботі з списками.</i></p> <p><b><u>Завдання на СРС</u></b> 1. Вивчення лекційного матеріалу. Вивчення рекомендованої літератури. Вивчення термінів і основних понять з теми. Підготовка до комп'ютерного практикуму і розробка звіту з лабораторної роботи [1 - 3, 5 - 7]</p>
10	<p><b><i>Поняття "процедура", "функція", "функціональне програмування" та "процедурно-орієнтоване" програмування</i></b> <i>Поняття "процедура". Синтаксис об'явлення та виклику процедур. Особливості використання глобальних та локальних імен в Python.</i></p> <p><b><u>Завдання на СРС</u></b> 1. Вивчення лекційного матеріалу. Вивчення рекомендованої літератури. Вивчення термінів і основних понять з теми. Підготовка до комп'ютерного практикуму і розробка звіту з лабораторної роботи [1 - 3, 5 - 7]</p>
11	<p><b><i>Поняття "процедура", "функція", "функціональне програмування" та "процедурно-орієнтоване" програмування</i></b></p>



№ з/п	Назва теми лекції та перелік основних питань (перелік дидактичних засобів, завдання на СРС з посиланням на літературу)
	<p><i>Розміщення аргументів по порядку та по імені при виклику процедури, завдання значення аргументу "по замовчуванню". Аргументи функції. Змінне число параметрів</i></p> <p><b><u>Завдання на СРС</u></b></p> <p>1. Вивчення лекційного матеріалу. Вивчення рекомендованої літератури. Вивчення термінів і основних понять з теми. Підготовка до комп'ютерного практикуму і розробка звіту з лабораторної роботи [1 - 3, 5 - 7]</p>
12	<p><b><i>Базові поняття об'єктно - орієнтованого програмування</i></b> <i>Особливості об'єктно - орієнтованого підходу в середовищі Python: відмінності та схожість з іншими системами програмування. Діаграми класів UML. Позначення атрибутів екземплярів, методів екземплярів, основні поняття наслідування.</i></p> <p><b><u>Завдання на СРС</u></b></p> <p>1. Вивчення лекційного матеріалу. Вивчення рекомендованої літератури. Вивчення термінів і основних понять з теми. Підготовка до комп'ютерного практикуму і розробка звіту з лабораторної роботи [1 - 3, 5 - 7]</p> <p>2. <b>Тема 1.7.</b> Процедури, функції та модульність в Python. <i>Особливості об'єктно - орієнтованого підходу в середовищі Python: відмінності та схожість з іншими системами програмування.</i> <i>Діаграми класів UML.</i> <i>Позначення атрибутів екземплярів, методів екземплярів, основні поняття наслідування.</i> <i>Особливості та додаткові можливості ООП в Python.</i> <i>Емпіричні вимірювання ефективності алгоритмів. Накладні витрати алгоритмів за часом і пам'яттю. Поняття складності обчислення. Правила обчислення часу виконання.</i> <i>Побудова віконної програми відображення бази даних, яка формує дочірні вікна різного типу (діалогові, модальні, незалежні), з різними варіантами розміщення елементів (мітка, кнопка, поле введення тексту).</i></p> <p><b>Література</b> основна [1 (ст.152-200)]; додаткова [4 (ст.611-891)]; інформаційні ресурси [7 (розділи 10-12)]</p>
13	<p><b><i>I. Особливості та додаткові можливості ООП в Python</i></b> <i>Особливості та додаткові можливості ООП в Python.</i></p> <p><b><u>Завдання на СРС</u></b></p> <p>1. Вивчення лекційного матеріалу. Вивчення рекомендованої літератури. Вивчення термінів і основних понять з теми. Підготовка до комп'ютерного практикуму і розробка звіту з лабораторної роботи [1 - 3, 5 - 7]</p> <p><b><i>II. Основи теорії обчислюваності</i></b> <i>Емпіричні вимірювання ефективності алгоритмів. Накладні витрати алгоритмів за часом і пам'яттю. Поняття складності обчислення. Правила обчислення часу виконання.</i></p> <p><b><u>Завдання на СРС</u></b></p> <p>1. Вивчення лекційного матеріалу. Вивчення рекомендованої літератури. Вивчення термінів і основних понять з теми. Підготовка до комп'ютерного практикуму і розробка звіту з лабораторної роботи [1 - 3, 5 - 7]</p>
14	<p><b><i>Файли. Обробка винятків</i></b> <i>Робота з файлами. Обробка винятків</i></p> <p><b><u>Завдання на СРС</u></b></p> <p>1. Вивчення лекційного матеріалу. Вивчення рекомендованої літератури. Вивчення термінів і основних понять з теми. Підготовка до комп'ютерного практикуму і розробка звіту з лабораторної роботи [1 - 3, 5 - 7]</p>
15	<p><b><i>I. Кібернетика, як метод пізнання</i></b> <b><i>Математичний аналіз. Фізичний експеримент.</i></b> <b>Література</b> основна [1 - 3]; додаткова [4] <b>Завдання на СРС</b> Вивчення лекційного матеріалу. Вивчення рекомендованої літератури. Вивчення термінів і основних понять з теми. Підготовка до комп'ютерного практикуму і розробка звіту з лабораторної роботи</p>

№ з/п	Назва теми лекції та перелік основних питань (перелік дидактичних засобів, завдання на СРС з посиланням на літературу)
	<p><b>II. Парадигми пізнання</b>  <b>Поняття інформації. Детермінізм.</b>  <b>Поняття інформації. Стохастичність</b>  <b>Література</b> основна [1 - 3]; додаткова [4]  <b>Завдання на СРС</b> Вивчення лекційного матеріалу. Вивчення рекомендованої літератури. Вивчення термінів і основних понять з теми. Підготовка до комп'ютерного практикуму і розробка звіту з лабораторної роботи</p>

## 5.2. Комп'ютерні практикуми

### Основні завдання циклу комп'ютерних практикумів:

- формування вмінь оптимального і ефективного застосування сучасних інформаційних технологій при проектуванні компонентів програмного забезпечення, набування здатностей до застосування теоретичних знань в практичній діяльності, опанування технікою розробки алгоритмів із застосуванням мови Python, самостійна розробка і створення програм, які реалізують різні засоби використання типів даних високого рівня.
- більш широко використовувати можливості інформаційного пакету програмного забезпечення Microsoft Office в подальшій навчальній та професійній діяльності студента

№ з/п	Назва теми заняття
1	<p><b>Внутрішнє машинне представлення цілих та дійсних чисел</b>  Очікується, що студенти засвоять наступні питання та застосують їх при виконанні завдання за темою ДКР  Правила переведення цілих та дійсних чисел з однієї системи числення в іншу</p>
2	<p><b>Побудова блок-схем із застосуванням середовища Visio</b>  Очікується, що студенти засвоять наступні питання та застосують їх при виконанні завдання за темою ДКР  Ознайомлення із правилами побудови блок-схем</p>
3	<p><b>Робота в інтегрованому середовищі розробки IDLE</b>  Очікується, що студенти засвоять наступні питання та застосують їх при виконанні завдання за темою ДКР  Знайомство із середовищем IDLE. Збереження коду. Розробка найпростіших програм</p>
4	<p><b>Типи даних. Поняття змінної. Введення даних з клавіатури</b>  Очікується, що студенти засвоять наступні питання та застосують їх при виконанні завдання за темою ДКР  Дані та їх типи. Цілі числа, числа з плаваючою точкою, рядки, логічний тип даних.</p>
5	<p><b>Програмування алгоритмів розгалуженої структури</b>  Очікується, що студенти засвоять наступні питання та застосують їх при виконанні завдання за темою ДКР  Логічні вирази. Логічні оператори. Особливості введення даних з клавіатури. Умовний оператор. Інструкція if..</p>
6	<p><b>Програмування алгоритмів циклічної структури</b>  Очікується, що студенти засвоять наступні питання та застосують їх при виконанні завдання за темою ДКР  Цикли. Універсальний організатор циклу - інструкція while.</p>
7	<p><b>Програмування алгоритмів циклічної структури</b>  Очікується, що студенти засвоять наступні питання та застосують їх при виконанні завдання за темою ДКР  Інструкція for.</p>
8	<b>Послідовності: списки.</b>

№ з/п	Назва теми заняття
	<i>Очікується, що студенти засвоять наступні питання та застосують їх при виконанні завдання за темою ДКР</i> <i>Поняття списку. Способи організації та принципи роботи зі списками.</i>
9	<b>Побудова програм формування одномірного списку (Ч-1)</b> <i>Очікується, що студенти засвоять наступні питання та застосують їх при виконанні завдання за темою ДКР</i> <i>Одномірні масиви. Формування одномірного списку псевдовипадкових цілих чисел та визначення загальних його характеристик: суми елементів, максимального та мінімального значень та середнього значення</i>
10	<b>Побудова програм формування одномірного списку (Ч-2)</b> <i>Очікується, що студенти засвоять наступні питання та застосують їх при виконанні завдання за темою ДКР</i> <i>Одномірні масиви. Формування одномірного списку псевдовипадкових цілих чисел та визначення загальних його характеристик: суми елементів, максимального та мінімального значень та середнього значення</i>
11	<b>Побудова програм формування двовимірного списку (Ч- 1)</b> <i>Очікується, що студенти засвоять наступні питання та застосують їх при виконанні завдання за темою ДКР</i> <i>Багатовимірні масиви. Динамічне виділення пам'яті під масиви.</i>
12	<b>Побудова програм формування двовимірного списку (Ч-2)</b> <i>Очікується, що студенти засвоять наступні питання та застосують їх при виконанні завдання за темою ДКР</i> <i>Формування двовимірного списку псевдовипадкових дійсних чисел та визначення загальних його характеристик: суми елементів, максимального та мінімального значень та середнього значення</i>
13	<b>Побудова програм з використанням строкового типу даних (Ч-1)</b> <i>Очікується, що студенти засвоять наступні питання та застосують їх при виконанні завдання за темою ДКР</i> <i>Поняття послідовності в мові Python. Складний тип даних рядок.</i>
14	<b>Побудова програм з використанням строкового типу даних (Ч-2)</b> <i>Очікується, що студенти засвоять наступні питання та застосують їх при виконанні завдання за темою ДКР</i> <i>. Особливості організації рядків. Робота з рядками.</i>
15	<b>Послідовності: кортеж, словник. (Ч-1)</b> <i>Очікується, що студенти засвоять наступні питання та застосують їх при виконанні завдання за темою ДКР</i> <i>Поняття кортежу і словника. Способи організації.</i>
16	<b>Послідовності: кортеж, словник. (Ч-2)</b> <i>Очікується, що студенти засвоять наступні питання та застосують їх при виконанні завдання за темою ДКР</i> <i>Принципи роботи із кортежами та словниками.</i>
17	<b>Функції. Параметри і аргументи функцій. Локальні і глобальні змінні (Ч-1)</b> <i>Очікується, що студенти засвоять наступні питання та застосують їх при виконанні завдання за темою ДКР</i> <i>Поняття функції. Опис функції. Виклик функції.</i>
18*	<b>Функції. Параметри і аргументи функцій. Локальні і глобальні змінні (Ч-2)</b> <i>Очікується, що студенти засвоять наступні питання та застосують їх при виконанні завдання за темою ДКР</i> <i>Поняття параметра та аргументу.</i>
19	<b>Побудова програм з використанням рекурсивних процедур та функцій (Ч-1)</b> <i>Очікується, що студенти засвоять наступні питання та застосують їх при виконанні завдання за темою ДКР</i> <i>Рекурсія. Прямая (безпосередня) рекурсія.</i>

№ з/п	Назва теми заняття
20*	<p><b>Побудова програм з використання рекурсивних процедур та функцій (Ч-2)</b>  <u>Очікується, що студенти засвоять наступні питання та застосують їх при виконанні завдання за темою ДКР</u>            Пряма (безпосередня) рекурсія. Непряма рекурсія</p>
21	<p><b>Побудова віконної програми (Ч-1)</b>  <u>Очікується, що студенти засвоять наступні питання та застосують їх при виконанні завдання за темою ДКР</u>            Побудова віконної програми відображення бази даних, яка формує дочірні вікна різного типу (діалогові, модальні, незалежні)</p>
22*	<p><b>Побудова віконної програми (Ч-2)</b>  <u>Очікується, що студенти засвоять наступні питання та застосують їх при виконанні завдання за темою ДКР</u>            Побудова віконної програми відображення бази даних, яка формує дочірні вікна різного типу (діалогові, модальні, незалежні), з різними варіантами розміщення елементів (мітка, кнопка, поле введення тексту).</p>
23	<p><b>Файли (Ч-1)</b>  <u>Очікується, що студенти засвоять наступні питання та застосують їх при виконанні завдання за темою ДКР</u>            Робота з бінарними фалами</p>
24	<p><b>Файли (Ч-2)</b>  <u>Очікується, що студенти засвоять наступні питання та застосують їх при виконанні завдання за темою ДКР</u>            Робота з бінарними фалами</p>
25	<b>МКР</b>
1	<p><b>Microsoft Office Word</b>  <u>Очікується, що студенти засвоять наступні питання та застосують їх при виконанні завдання за темою ДКР</u>            Вимоги до документу. Навігація. Стилi та заголовки. Зміст Вставки. Нумерація.</p>
2	<p><b>Microsoft Office Word</b>  <u>Очікується, що студенти засвоять наступні питання та застосують їх при виконанні завдання за темою ДКР</u>            Таблиці Рисунки Формули Заяви. Редакція тексту. Формування шаблонів звітів та ДКР. Перехресні посилання та список літератури.</p>
3	<p><b>Microsoft Office Excel (Частина 1)</b>  <u>Очікується, що студенти засвоять наступні питання та застосують їх при виконанні завдання за темою ДКР</u></p> <ol style="list-style-type: none"> <li>1. Форматування листів.</li> <li>2. Створення, заповнення, редагування та форматування таблиць</li> <li>3. Працювати з таблицями та списками із застосуванням логічних змінних, текстових та календарних функцій.</li> <li>4. Працювати з таблицями та списками із застосуванням математичних формул.</li> </ol>
4	<p><b>Microsoft Office Excel (Частина 2)</b>  <u>Очікується, що студенти засвоять наступні питання та застосують їх при виконанні завдання за темою ДКР</u></p> <ol style="list-style-type: none"> <li>1. Використання стандартних функцій процесора MS Excel для розробки біологічних моделей</li> <li>2. Робота зі списком, що випадає</li> <li>3. Побудова графіків в EXCEL</li> </ol>
5	<p><b>Microsoft Office Power Point</b>  <u>Очікується, що студенти засвоять наступні питання та застосують їх при виконанні завдання за темою ДКР</u>            Вимоги до оформлення презентацій. Види презентацій. Шаблони презентації</p>

№ з/п	Назва теми заняття
	Обговорення поширених помилок при формування презентацій. Формування презентації з захисту індивідуального завдання

### **Платформа дистанційного навчання:**

Для кращого засвоєння матеріалу навчальної дисципліни в період дистанційної роботи, використовується електронна пошта, платформа дистанційного навчання «Сікорський» на основі системи Google Classroom та платформа для проведення онлайн-зустрічей Google Meet та ZOOM, за допомогою яких:

- спрощується розміщення методичних рекомендацій, навчальних матеріалів, літератури тощо;
- здійснюється зворотній зв'язок зі студентами щодо навчальних завдань та змісту навчальної дисципліни;
- перевіряються і оцінюються виконані завдання;
- ведеться облік виконання студентами плану навчальної дисципліни, дотримання графіку подання навчальних/індивідуальних завдань та їх оцінювання

## **6. Самостійна робота студента**

Самостійна робота передбачає: підготовку до лекцій та комп'ютерних практикумів; до участі в обговоренні питань відповідної теми; самоконтроль набутих знань; опрацювання рекомендованих джерел та літератури; підготовку до виконання модульної контрольної роботи та індивідуального завдання; екзамену тощо.

6.1. Теми для самостійного опрацювання – не заплановано

6.2. Підготовка до лекційних та комп'ютерних практикумів. Для підготовки до лекційних та комп'ютерних практикумів студенту необхідно опрацювати заплановану базову та допоміжну літературу, рекомендовані джерела та підготувати матеріал для його обговорення та виконання на заняттях. На це студенту виділяється по 1-1,5 години на кожну тему дисципліни.

6.3. Модульна контрольна робота. На підготовку до МКР відводиться 2-3 години СРС. Перелік питань для підготовки МКР надано у **Додатку В**.

6.4. Домашня контрольна робота (ДКР). На підготовку та оформлення ДКР відводиться 10 годин СР. Тему ДКР студент повинен обрати та ухвалити у викладача не пізніше 4 тижня від початку занять. Рекомендований список тем ДКР надано у **додатку Д**. Терміни подання ДКР викладачу не пізніше 16 тижня. Захист ДКР планується на позаплановому занятті в термін з 17 по 18 тиждень.

6.5. Екзамен. Екзамен проводиться в період екзаменаційної сесії, по завершенню навчального семестру згідно ухваленого графіку. На підготовку до екзамену відводиться 30годин СР. Перелік питань для підготовки до екзамену надано у **додатку А**. В період дистанційного навчання екзамен може бути проведений згідно графіку за допомогою Google Classroom та платформи для проведення онлайн-зустрічей Google Meet.

## **4. Політика та контроль**

### **7. Політика навчальної дисципліни (освітнього компонента)**

Студентам рекомендується дотримуватись правил відвідування занять та поведінки на них.

#### **7.1. Правила відвідування занять:**

Лекції. Сьогодні практично немає нових україномовних (або зарубіжних) навчальних видань, цілковито присвячених цій проблематиці. Тому для студентів дуже важливо відвідувати

лекції, на яких висвітлюватиметься систематизований навчальний матеріал, презентації текстів документів в обсязі достатньому для опанування навчальної дисципліни студентом.

Відвідування лекцій для студентів з даної навчальної дисципліни є обов'язковим, адже саме на лекціях вони отримують необхідні базові теоретичні знання, що дає їм змогу ґрунтовно підготуватися до комп'ютерних практикумів, модульної контрольної роботи, ДКР та заліку. Нового навчального матеріалу з навчальної дисципліни, крім матеріалу лекцій, вкрай мало а той що є застарілий та не відповідає сьогоденню. Тому для студентів, які бажають продемонструвати відмінні результати навчання, активна робота на лекційних заняттях просто необхідна. Однак відпрацьовувати пропущені лекції не потрібно

*Комп'ютерні практикуми* Активна участь студента на комп'ютерних практикумах (надалі – заняттях) є обов'язковою і буде вимагатись. Рейтинг студента значною мірою формуватиметься за результатами його роботи на цих заняттях. Кожне пропущене заняття (незалежно від причин пропуску) знижує підсумковий рейтинг студента з навчальної дисципліни. Немає конкретної кількості пропущених занять, які потребуватимуть самостійного опрацювання студентом відповідних тем і додаткового спілкування з цього приводу з викладачем. Разом з тим, студент, який пропустив заняття, може отримати низький рейтинг, який не дозволить допустити такого студента до заліку. В такому разі теми з пропущених занять мають бути обов'язково вивчені самостійно, а студент обов'язково виконані звіти з відповідної теми та відповідно захищені.

## 7.2. Правила виконання завдань:

Опрацьовуючи навчальний матеріал навчальної дисципліни «Основи інформатики», студенти:

- 1) самостійно:
  - готуються до МКР;
  - готуються до занять та виконання звітів;
  - виконують ДКР;
  - оформлюють звіти та ДКР у відповідності до вимог їх оформлення.
- 2) на заняттях:
  - беруть участь у дискусіях, інтерактивних формах організації заняття;
  - своєчасно виконують МКР та індивідуального завдання у вигляді ДКР;
  - виконують та захищають звіти з комп'ютерних практикумів (надалі – звіти).

Теми, завдання для занять, МКР, ДКР передбачені програмою навчальної дисципліни, доступні в особистому кабінеті студента в системі «Кампус», Google-класі на платформі Сікорський з даної навчальної дисципліни тощо.

Завдання та матеріали для проведення модульної контрольної роботи та домашньої контрольної роботи подаються у додатку до робочої навчальної програми (силлабусу).

## 7.3. Правила поведінки на заняттях

**Тематика лекцій** висвітлена у силлабусі. Структурування лекційного матеріалу спрямовано на формування такої системи подачі лекції, що передбачає визначення критеріїв наукових понять, явищ, процесів, логічно пов'язаних, об'єднаних загальною темою

При розгляді основних питань лекції викладачем акцентується увага на формуванні у студентів логічного мислення, розуміння побудови логічних висновків, уміння використання мови Python..

На лекції допускається діалог між студентами та викладачем.

**Тематика комп'ютерних практикумів** висвітлена у силлабусі Під час проведення занять рекомендується організація дискусії між студентами з проблемних питань курсу, спрямовування цих дискусій до формування власної позиції та оцінки студентом викладеного матеріалу.

Під час занять викладач узагальнюватиме та аналізуватиме помилки і недоліки складених студентами доповідей/роботи (особистої точки зору), відповідатиме на питання студентів, а студенти оцінюватимуть доповіді один одного, доповнюючи або виявляючи недоліки в доповіді/роботі.

Очікується, що на занятті кожен студент бере активну участь в обговоренні тематики занять. Очікується, що студенти пояснюватимуть, чому вони сформували власну думку саме таким чином, а також висловлюватимуть свою думку щодо думок інших студентів.

Студенти, які пропустили заняття, повинні не допускати зниження підсумкового рейтингу, своєчасно (протягом семестру) опрацювавши відповідні теми і виконувати завдання, передбачені для пропущених занять чи завдань.

На лекціях та заняттях допускається використання ноутбуків, смартфонів, але лише для цілей, зумовлених темою заняття і відповідним тематичним завданням. Використовувати зазначені (та інші подібні) засоби для розваги чи спілкування під час заняття не варто. Відповідати на питання викладача, читаючи з екрану смартфона, ноутбуку чи з підручника не варто також. Це характеризує рівень підготовки студента не з кращого боку.

Студент на занятті може використовувати підготовлені ним письмові нотатки з питань теми заняття (або передбачених завданням), однак висловлювати позицію, читаючи з аркуша паперу не варто. Це також характеризує рівень підготовки студента не з кращого боку.

Не рекомендується використовувати документи, знайдені з ненадійних джерел мережі інтернет (не рекомендованих викладачем), оскільки, як правило, вони мають низьку якість та сформовані з використанням застарілого інформаційного матеріалу, або недостовірної інформації з певного питання.

#### **7.4. Правила виконання індивідуального завдання**

Студенти повинні не пізніше 3-4 тижня отримати та ухвалити теми ДКР.

Розділи ДКР повинні виконуватись протягом семестру. Не пізніше 16-17 тижня студент повинен надати на перевірку ДКР на перевірку виконання індивідуального завдання та оформлення роботи до відповідних вимог оформлення.

На захист ДКР студенти надають презентацію.

В рейтинг оцінювання ДКР входить наступні компоненти:

- правильність виконання індивідуального завдання;
- оформлення ДКР;
- презентація;
- захист ДКР.

#### Основні вимоги до складових ДКР

Зміст завдання полягає в створенні кожним студентом процедур на мові програмування Python, що містять алгоритми сортування, злиття та пошуку, алгоритми роботи з графами та деревами та рекурсивні алгоритми. Крім коду програми кожному студенту треба надати пояснювальну записку з розрахунком часової складності та блок-схему, а також результати тестування для різних вхідних параметрів

Теми ДКР надані в **Додаток Д** до силабусу.

Вимоги до оформлення ДКР та презентації до неї надані в посібнику з навчальної дисципліни «*Основи програмування та інформатики: домашня контрольна роботи*» та розглядаються на комп'ютерних практикумах з розділу 2 «*Основи інформатики*» навчальної дисципліни.

#### **7.5 Заохочувальні та штрафні бали**

##### Заохочувальні бали

- |   |          |
|---|----------|
| 1) участь у факультетській олімпіаді з дисципліни                         | 6 балів; |
| 2) модернізації практичних робіт  | 6 балів; |
| 3) виконання завдань із удосконалення дидактичних матеріалів з дисципліни | 6 балів. |

##### Штрафні бали

- 1) несвоєчасне (пізніше ніж на тиждень) подання звіту з комп'ютерного практикуму  
- 1 бал (за кожен звіт)
- 2) За кожний тиждень запізнення з поданням ДКР на перевірку нараховується штрафний – 1 бал (усього не більше –5 балів)

## 7.6. Політика університету

### Академічна доброчесність.

Політика та принципи академічної доброчесності визначені у розділі 3 Кодексу честі Національного технічного університету України «Київський політехнічний інститут імені Ігоря Сікорського». Детальніше: <https://kpi.ua/code>

### Норми етичної поведінки.

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>

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

**Поточний контроль:** здійснюється під час навчальних занять і має на меті перевірити рівень підготовки студентів до навчальних занять. Під час комп'ютерних практикумів проводиться виконання та захист 11 комп'ютерних звітів з розділу 1 та виконання (без захисту) 4 комп'ютерних звітів з розділу 2, з акцентуванням на узагальнені й науковій інформації, рекомендації до їх виконання та оформлення, формулювання власної позиції та оцінки викладеного / проаналізованого матеріалу. Модульна контрольна робота проводиться в кінці семестру, після закінчення викладання теоретичного матеріалу. Виконання та захист індивідуального завдання (домашньої контрольної роботи).

**Календарний контроль:** проводиться двічі на семестр як моніторинг поточного стану виконання вимог силлабусу. Є два можливих результати календарного контролю: атестований (а) та неатестований (н/а). Результат залежить від кількості набраних балів на момент проведення календарного контролю відповідно до вимог КПП ім. Ігоря Сікорського

**Семестровий контроль:** **екзамен**

### Оцінювання та контрольні заходи

Рейтинг студента з кредитного модуля складається з балів, що отримуються за:

- 1) виконання та захист 11 звітів з розділу 1 та виконання (без захисту) 4 звітів з розділу 2;
- 2) модульна контрольна робота
- 3) ДКР
- 4) екзаменаційна робота

Робота студента:

- на заняттях (виконання та захист звітів) визначає 30% його рейтингу з кредитного модуля;

- модульна контрольна робота (надалі МКР) визначає 15% його рейтингу з кредитного модуля;

- ДКР визначає 15% його рейтингу з кредитного модуля

- екзаменаційна робота визначає 40% його рейтингу з кредитного модуля .

Студент отримає найвищий рейтинг, якщо він:

- своєчасно виконує та захищає звіти. Звіти оформлює до відповідних вимог викладача.
- бере активну участь на заняттях, переважно надає повні та аргументовані відповіді, логічно їх викладає, висловлює власну позицію з питань занять, дану позицію викладає чітко і логічно, обґрунтовує її належним чином а також активно доповнює відповіді інших студентів на занятті;
- своєчасно виконує та оформить за відповідними вимогами викладача ДКР
- своєчасно готується та виконує МКР (завдання до МКР надано в **додатку В** до силлабусу). Студенту надається одноразова можливість написати МКР.



Для того щоб отримати найвищий рейтинг, студенту потрібно: своєчасно виконувати та захищати звіти з комп'ютерних практикумів; своєчасно виконати МКР; своєчасно виконати та захистити ДКР відповідно до вимог її оформлення.

До зниження рейтингу студентів призводить: невиконання звітів, МРК та ДКР; неточності, неповнота, помилки у відповідях чи ґрунтуваннях на недостовірних інформаційних джерел при захисті звітів та ДКР.

Викладач оцінює роботу студента на кожному занятті за результатами наданих звітів та їх захисту або за результатом виконання індивідуального завдання (ДКР), але конкретна підсумкова кількість балів за роботу на заняттях виставляється викладачем під час першого і другого етапу проміжної атестації – на восьмому і шістнадцятому тижнях навчання відповідно. Рейтинг студента станом на 8-й тиждень (за результатами роботи на 2-5 заняттях або 2-х звітів) і 16-й тиждень (за результатами роботи на наступних 14-15 заняттях або 8 звітів). Результати навчання (календарний контроль, МКР, звіти, ДКР) відображаються в модулі «Поточний контроль» Електронного кампусу, Google-класі тощо.

#### **Процедура оскарження результатів контрольних заходів**

Деталізовані критерії оцінювання результатів навчання студента визначені в положенні про РСО з навчальної дисципліни, що є у **Додатку С** до силабусу.

Студент може оскаржити оцінку викладача, подавши відповідну скаргу викладачу не пізніше, ніж наступного дня після ознайомлення студента з виставленою оцінкою. Скарга розглядатиметься за процедурами, встановленими в університеті.

**Умови допуску до семестрового контролю:** *Наявність кількості балів не менше 30, виконання модульної контрольної роботи не менше ніж на «достатньо», виконання та захист всіх звітів не менше ніж на «задовільно» а також виконання та захист ДКР не менше ніж на «достатньо»*

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
Не виконані умови допуску	Не допущено

#### **9. Додаткова інформація з дисципліни (освітнього компонента)**

Перелік питань до семестрового контролю (екзамену) надані в **додатку А** до силабусу.

При наявності у студенту документів підтверджуючих його участь у олімпіадах (міських, міжміських, Всеукраїнських тощо) за темою заняття або розділу кредитного модуля можуть зараховуватись за відповідною тематикою та відповідними балами РСО

#### **Рекомендації студентам**

Працюючи на лекції, студенту варто конспектувати основні поняття, ознаки, принципи, класифікації, визначення, алгоритми, про які розповідатиме викладач. Якщо студент буде уважно слухати, фіксувати відповідний матеріал, він потім зможе прочитати цей текст та застосувати його при вирішенні завдання чи підготовці до практичного заняття. Якщо після цього студент презентує свою обґрунтовану позицію (думку), критично оцінить позиції (думки) інших студентів, ставитиме питання викладачу та студентам – обсяг засвоєного ним навчального матеріалу і глибина його розуміння збільшуватиметься в рази.

Готуючись до заняття студент має обов'язково опрацювати лекційний матеріал певної теми, бажано ознайомитись з додатковими ресурсами в мережі. При виникненні питань, виявленні незрозумілих положень, необхідно обов'язково обговорити їх з викладачем. На занятті навіть

добре підготовлений студент не повинен залишатись пасивним спостерігачем, а активно включатись у обговорення питання або захисту звіту. Якщо ж студент не ознайомився з навчальним матеріалом, йому варто уважніше слухати виступаючих, і завдяки отриманій інформації намагатись компенсувати недоліки підготовки до заняття. Не слід відмовляти від відповіді на питання викладача. Навіть якщо студент не знає відповіді, доцільно спробувати відповісти, висловити свою думку, виходячи з власних знань, досвіду, логіки запитання тощо. При цьому не треба боятись помилитися: одним з важливих завдань вивчення кредитного модуля є вироблення вміння логічно мислити, вирішувати поставлені задачі і відповідно висловлювати власні думки. Однак, варто пам'ятати, що незнання матеріалу кредитного модуля є суттєвим недоліком роботи студента і буде негативно впливати на його загальний рейтинг. Відповідальне ставлення до підготовки на кожне заняття дає змогу не лише правильно засвоїти навчальний матеріал, але й зекономити зусилля при проходженні семестрового контролю.

Важливим у належній підготовці студента є вироблення в нього вміння працювати з документами, які мають практичне значення в вирішенні поставлених задач.

Ознайомлюючись із новим для себе інформаційним документом/джерелом, слід, насамперед намагатись виявити його достовірність, зрозуміти логіку та послідовність викладеного матеріалу. Такий аналіз дозволить студенту не лише краще засвоїти інформацію, алей аналізувати послідовність виконання дій в розробці програмного продукту (надалі ПП).

У разі складнощів з розумінням деяких послідовностей розробки ПП не варто соромитись і необхідно обов'язково звертатись до викладача. Він обов'язково допоможе.

### **Позааудиторні заняття**

Можлива участь студентів в щорічних галузевих виставок «Охорона здоров'я», а також профільних семінарів, наукових конференцій, олімпіадах тощо

### **Дистанційне навчання**

Можливе синхронне та асинхронне дистанційне навчання з використанням платформ для відео-конференцій (Google Meet, Microsoft Teams, Zoom, Skype тощо) та освітньої платформи дистанційного навчання «Сікорський» (Moodle, Google Classroom).

### **Інклюзивне навчання**

Допускається

### **Робочу програму навчальної дисципліни (силлабус):**

#### **Складено**

доц. каф. БМК, доц., к.ф.-м..н., Федорін Ілля Валерійович, (посада, науковий ступінь, вчене звання, прізвище, ім'я, по батькові)
ст.вик. каф. БМК Корнієнко Галина Альбертівна

Ухвалено кафедрою біомедичної кібернетики (протокол № 18 від 24 06 2024 р.)

Погоджено Методичною комісією факультету ФБМІ<sup>1</sup> (протокол № 9 від 26 06 2024 р.)

<sup>1</sup> Методичною радою університету – для загальноуніверситетських дисциплін.

Перелік питань до семестрового контролю (екзамен)

Зразок екзаменаційного білету

(Форма N Н-5.04)

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ  
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ ім. ІГОРЯ СІКОРСЬКОГО»

Рівень вищої освіти першого (бакалаврського)  
(назва ступеня)

Спеціальність 122 Комп'ютерні науки  
(код і назва напрямку підготовки)

Освітня програма Комп'ютерні технології в біології та медицині  
(код і назва спеціальності)

Навчальна дисципліна Основи інформатики та програмування  
(назва)

ЕКЗАМЕНАЦІЙНИЙ БІЛЕТ № \_\_\_\_\_

- 1 Питання з I блоку питань
- 2 Практичне завдання II блок
- 3 Практичне завдання III блок

Затверджено на засіданні кафедри Біомедичної кібернетики  
(назва кафедри)

Протокол № \_\_\_\_\_ від « \_\_\_\_\_ » \_\_\_\_\_ 20 \_\_\_\_\_ р.

Завідувач кафедри \_\_\_\_\_  
(підпис) Ім'я ПРИЗВИЩЕ  
(Прізвище та ініціали)

**ПИТАННЯ для формування екзаменаційних білетів \***

**Питання I з блоку питань**

1. Структура програми на мові Python. Синтаксис опису функції
2. Поняття алгоритму та його властивості. Види алгоритмів і її зображення в блок-схемах.
3. Класичні алгоритми програмування (підрахунок суми, добутку, кількості, знаходження максимуму послідовності, організація пошуку даних).
4. Поняття змінної та її властивості. Синтаксис об'яви змінної. Правила роботи оператора присвоювання. Форми запису.
5. Основні типи даних Python. Функції перетворення типів. Перетворення результатів розрахунку арифметичних виразів.
6. Аналіз умов. Синтаксичні конструкції та правила роботи оператора if, else, elif. Правила логічних операцій. Правила розрахунку логічних операцій.
7. Керування пам'яттю у Python. Виділення та вивільнення пам'яті.
8. Циклічні конструкції Python. Синтаксис та правила роботи операторів for, while. Умова продовження роботи циклу.
9. Контейнери Python. Загальна характеристика контейнерів.
10. Списки та кортежі у Python. Загальна характеристика, властивості, базові методи списків. Правила використання пам'яті при роботі за списками.
11. Словники та множини у Python. Загальна характеристика словників. Базові методи словників.
12. Строки у Python. Загальна характеристика строк та базові методи.
13. Основи ООП у Python. Поняття класу, об'єкту, методу класа, властивості класів.

14. Декоратори у Python. Властивості та приклади реалізації. Шаблон Singleton.  
 15. Основи структурного програмування. Створення підпрограм. Прототипи функцій. Формальні параметри та аргументи. Виклик функцій.

## Питання II ПРАКТИЧНА ЗАДАЧА I

1. Напишіть програму Python для друку всіх унікальних значень у словнику.

Приклад входу : [{"V": "S001"}, {"V": "S002"}, {"VI": "S001"}, {"VI": "S005"}, {"VII": "S005"}, {"V": "S009"}, {"VIII": "S007"}]  
 Результат : Unique Values: {'S005', 'S002', 'S007', 'S001', 'S009'}

2. Напишіть програму Python, щоб поєднати два словники, додавши значення для однакових ключів.

d1 = {'a': 100, 'b': 200, 'c': 300}  
 d2 = {'a': 300, 'b': 200, 'd': 400}  
 Результат: {'a': 400, 'b': 400, 'd': 400, 'c': 300}

3. Напишіть програму Python, щоб створити словник із рядка.

Примітка: відстежте кількість літер з рядка.  
 Вхід: 'w3resource'  
 Результат: {'3': 1, 's': 1, 'r': 2, 'u': 1, 'w': 1, 'c': 1, 'e': 2, 'o': 1}

4. Напишіть програму Python для підрахунку значень, пов'язаних з ключем у словнику.

Вхід: = [{'id': 1, 'success': True, 'name': 'Lary'}, {'id': 2, 'success': False, 'name': 'Rabi'}, {'id': 3, 'success': True, 'name': 'Alex'}]  
 Результат: Порахуйте, скільки словників має 'success' - True

5. Напишіть програму Python, щоб отримати три найкращі товари в магазині.

Вхід: {'item1': 45.50, 'item2': 35, 'item3': 41.30, 'item4': 55, 'item5': 24}  
 Результат:  
 item4 55  
 item1 45.5  
 item3 41.3

6. Напишіть програму Python, щоб створити словник із двох списків, не втрачаючи повторюваних значень.

Вхід: ['Class-V', 'Class-VI', 'Class-VII', 'Class-VIII'], [1, 2, 2, 3]  
 Результат: {'Class-VII': 2, 'Class-VI': 2, 'Class-VIII': 3, 'Class-V': 1}

7. Напишіть програму Python, щоб відповідати ключовим значенням у двох словниках.

Вхід: {'key1': 1, 'key2': 3, 'key3': 2}, {'key1': 1, 'key2': 2}  
 Результат: key1: 1 є входить до x та y

8. Напишіть програму Python для створення та друку словника, який містить число (від 1 до n) у формі (x, x\*x).

Вхід ( n = 5 ) :  
 Результат: {1: 1, 2: 4, 3: 9, 4: 16, 5: 25}

9. Напишіть програму Python, щоб об'єднати наступні словники для створення нового.

Вхід:  
 dic1={1:10, 2:20}  
 dic2={3:30, 4:40}  
 dic3={5:50,6:60}  
 Результат: {1: 10, 2: 20, 3: 30, 4: 40, 5: 50, 6: 60}

10. Напишіть програму Python, щоб знайти список у списку списків, сума елементів яких найбільша.  
*Вхід:* [1,2,3], [4,5,6], [10,11,12], [7,8,9]  
*Результат:* [10, 11, 12]
11. Напишіть програму Python, щоб перевірити, чи всі словники у списку порожні чи ні.  
*Вхід:* [{},{},{}]  
*Результат:* True  
*Вхід:* [{1,2},{},{}]  
*Результат:* False
12. Напишіть програму Python для друку чотирьох значень десяткових, восьмеричних, шістнадцяткових (з великої літери), двійкових у одному рядку заданого цілого числа.  
*Вхід:* 25  
*Результат:*  
 Decimal Octal Hexadecimal (capitalized), Binary  
 25 31 19 11001
13. Напишіть клас Python, щоб знайти пару елементів (індекси двох чисел) із заданого масиву, сума якого дорівнює конкретному цільовому номеру.  
*Вхід:* numbers= [10,20,10,40,50,60,70], target=50  
*Результат:* 3, 4
14. Напишіть клас Python, щоб перевернути рядок слово за словом.  
*Вхід:* 'hello .py'  
*Результат:* '.py hello'
15. Напишіть програму Python для побудови наступного шаблону, використовуючи вкладений цикл.
- ```

*

* *

* * *

* * * *

* * * * *

* * * *

* * *

* *

*

```

### **Питання III ПРАКТИЧНА ЗАДАЧА II**

1. Напишіть програму Python для підрахунку кількості парних та непарних чисел із ряду чисел.  
*Вхід :* numbers = (1, 2, 3, 4, 5, 6, 7, 8, 9)  
*Результат:*

Number of even numbers : 5  
 Number of odd numbers : 4

2. Напишіть програму Python для друку букви 'X'.

Результат:

```
* *
* *
* *
*
* *
* *
* *
```

3. Напишіть програму Python для друку букви 'Z'.

Результат:

```
*****
*
*
*
*
*
*****
```

4. Напишіть програму Python, щоб знайти медіану з трьох значень.

Результат:

```
1 число: 15
2 число: 26
3 число: 29
Результат 26.0
```

5. Напишіть клас Python для реалізації pow(x, n).

Вхід: (2,-3), (3, 5), (100, 0)

Результат: 0.125, 243, 1

6. Напишіть програму Python для побудови наступного шаблону, використовуючи вкладений цикл.

Результат:

```
1
22
333
4444
55555
666666
7777777
88888888
999999999
```

7. Напишіть функцію Python, яка приймає рядок та обчислює кількість малих і великих літер.

Вхід: 'The quick Brow Fox'

Результат:

```
No. of Upper case characters : 3
No. of Lower case Characters : 12
```

8. Напишіть функцію Python, яка бере список і повертає новий список з унікальними елементами першого списку.  
Вхід: [1,2,3,3,3,3,4,5]  
Результат : [1, 2, 3, 4, 5]
9. Напишіть програму Python, щоб сортувати кортеж за його дробовим елементом.  
Вхід: [('item1', '12.20'), ('item2', '15.10'), ('item3', '24.5')]  
Результат: [('item3', '24.5'), ('item2', '15.10'), ('item1', '12.20')]
10. Напишіть програму Python, щоб видалити порожній кортеж із списку кортежів.  
Вхід: [( ), ( ), (" ), ('a', 'b'), ('a', 'b', 'c'), ('d')]  
Результат: [(" ), ('a', 'b'), ('a', 'b', 'c'), ('d')]
11. Напишіть програму Python, щоб підрахувати кількість символів (частоту символів) у рядку.  
Вхід: google.com'  
Результат: {'o': 3, 'g': 2, '.': 1, 'e': 1, 'l': 1, 'm': 1, 'c': 1}
12. Напишіть програму Python, щоб знайти першу появу підстроки 'not' та 'poor' у даній строці, якщо 'not' слідує за 'poor', замініть всю 'not...'poor' підстроку на 'good'. Поверніть результуючу строку.  
Вхід: 'The lyrics is not that poor!'  
'The lyrics is poor!'  
Результат: 'The lyrics is good!'  
'The lyrics is poor!'
13. Напишіть програму Python для підрахунку символів, що повторюються у рядку.  
Вхід: 'thequickbrownfoxjumpsoverthelazydog'  
Результат:  
o 4  
e 3  
u 2  
h 2  
r 2  
t 2
14. Напишіть клас Python для перетворення римської цифри в ціле число.  
Вхід: 'MMMCMLXXXVI', 'MMMM', 'C', 'IV'  
Результат: 3986, 4000, 100, 4
15. Напишіть клас Python, щоб перетворити ціле число в римське число.  
Вхід: 3986, 4000, 100, 4  
Результат: 'MMMCMLXXXVI', 'MMMM', 'C', 'IV'  
числа = 1000, 900, 500, 400, 100, 90, 50, 40, 10, 9, 5, 4, 1  
символи = "M", "CM", "D", "CD", "C", "XC", "L", "XL", "X", "IX", "V", "IV", "I"

## МОДУЛЬНА КОНТРОЛЬНА РОБОТА

Питання для проведення модульного контролю:

1. Яким буде значення змінної `smallest` після того, як даний код буде виконаний?

```
smallest = -1  
for num in [9, 41, 12, 3, 74, 15] :  
    if num < smallest:  
        smallest = num  
print(smallest)
```

3

-1

0

немає правильної відповіді

2. Що робить дана функція?

```
for i in range(10):  
    if not i%2 == 0:  
        print(i+1)
```

виводить непарні числа від 1 до 9

виводить парні числа від 0 до 8

виводить всі парні числа від 2 до 10

немає правильної відповіді

3. Що буде виведено?

```
>>> 7 % (5 // 2)
```

1

0

7

немає правильної відповіді

4. Що буде виведено?

```
y = 5  
y += 3  
y = int(str(y) + '2')  
print(y)
```

82

10

7

синтаксична помилка

5. Що буде виведено?

```
x = 4
```



```

y = 2
if not 1 + 1 == y or x == 4 and 7 == 8:
print('Yes')
elif x > y:
print('No')

```

'Yes'

'No'

'Yes'

'No'

нічого не буде виведено

6. Що буде виведено?

```

lst = [1, 2, 3, None, (), [], [[]], [()], {}, 4, 5, ]
print(len(lst))

```

11

12

немає правильної відповіді

синтаксична помилка

7. Що буде виведено?

```

nums = (55, 44, 33, 22)
print(max (min (nums[:2]), abs(-42) ) )

```

44

55

22

немає правильної відповіді

8. Скільки разів буде виконано тіло циклу?

```

n = 0
while n > 0 :
print('Loop')
print('Finish!')

```

жодного

це нескінченний цикл

4 рази

немає правильної відповіді

9. Який з наведених нижче кортежів буде більший за кортеж t?

t = (5, 1, 3)

(5, 2, 1)

(4, 0, 2)

(1, 10000, 7)

немає правильної відповіді

10. Що буде виведено?

```
fruit = 'Banana'
fruit[0] = 'b'
print(fruit)
```

banana  
bBanana  
виникне помилка  
немає правильної відповіді

11. Що буде виведено?

```
a = [1, 2, 3]
b = [4, 5, 6]
c = a + b
print(c)
```

[1, 2, 3, 4, 5, 6]  
[5, 7, 9]  
21  
виникне помилка

12. Розставте рядки в правильному порядку, щоб отримати функцію, яка обчислює суму всіх чисел в діапазоні від нуля до числа її аргумента (x)

```
def summa(x):
    res = 0
    for i in range(x):
        res += 1
    return res
```

13. Що буде виведено?

```
def foo(a, b):
    print(a, b)
foo(b = 1, a = 2)
```

2 1  
1 2  
нічого не буде виведено  
виникне помилка

14. Що буде виведено?

```
lst = [1, 2, 3, None, (), [], [[]], [()], {}, 4, 5, ]
print(type(lst[2]))
```

class 'int'  
class 'number'  
class 'list'  
class 'tuple'  
синтаксична помилка

15. Що з наведеного нижче є НЕдопустимою операцією для рядків:

```
s.append('\n')
s1 + s2
s * 4
s.index('k')
```

16. Яка з операцій витягнення зрізу зі списку

```
lst = [9, 41, 12, 3, 74, 15]
```

дасть список [12, 3]

```
lst[2:4]
```

```
lst[1:4]
```

```
lst[2:3]
```

немає правильної відповіді

17. В яких випадках Python розглядає об'єкт як "істину" (True)?

Істиною є ненульове число, або будь-який непустий об'єкт

Істиною є тільки булівська змінна, що дорівнює True

Ніякий об'єкт не може розглядатися як істина, істиною може бути тільки результат логічного виразу

немає правильної відповіді

18. Що буде виведено?

```
def foo(a, b=4, c=5):
```

```
print(a, b, c)
```

```
foo(1, 2)
```

```
1 2 5
```

```
1 4 5
```

```
1 2 4 5
```

немає правильної відповіді

19. Які з наведених нижче методів можуть бути застосовані як для списків, так і для кортежів?

```
count
```

```
index
```

```
append
```

```
sort
```

20. Що буде виведено?

```
a, b = 0, 1
```

```
while b < 5:
```

```
print(b, end=' ')
```

```
a, b = b, a+b
```

```
1 1 2 3
```

```
1 2 4
```

```
0 1 2 3 4
```

немає правильної відповіді

**РЕЙТИНГОВА СИСТЕМА ОЦІНЮВАННЯ РЕЗУЛЬТАТІВ НАВЧАННЯ**

з кредитного модуля

**ОСНОВИ ІНФОРМАТИКИ ТА ПРОГРАМУВАННЯ**

першого (бакалаврського) рівня вищої освіти ступеня «бакалавр»

форма навчання

денна

1. Рейтинг студента з кредитного модуля складається з балів, що він отримує за:

- виконання та захист 15 комп'ютерних звітів. (11 звітів за розділом 1 та 4 звіти за розділом 2)
- виконання модульної контрольної роботи
- домашня контрольна робота
- екзаменаційна робота

2. Критерії нарахування балів.

**2.1. Виконання та захист комп'ютерних практикумів**

Виконання та захист звіту студента оцінюється у 2 бали за такими критеріями:

|                                                                                                                                                                                                                                                                                                                                                 |             |
|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------|
| <ul style="list-style-type: none"> <li>● Робота виконана безпомилково, в повному обсязі, при захисті продемонстровані повні і міцні знання відповідного матеріалу</li> </ul> <b>Звіт</b> – надано своєчасно та дотримано усіх вимог по його оформленню.                                                                                         | - 2 бали    |
| <ul style="list-style-type: none"> <li>● В роботі допущені несуттєві неточності, при захисті продемонстровані знання відповідного матеріалу з несуттєвими неточностями</li> </ul> <b>Звіт</b> – надано своєчасно та дотримано усіх вимог по його оформленню                                                                                     | - 1,5 балів |
| <ul style="list-style-type: none"> <li>● Робота містить деякі помилки, які допущені через недбалість і відсутність сталих навичок, при захисті відповідного матеріалу відповідь студента неповна або містить неточну відповідь на теоретичні питання</li> </ul> <b>Звіт</b> – надано не своєчасно та не дотримано усіх вимог по його оформленню | - 1 бал     |
| <ul style="list-style-type: none"> <li>● В роботі допущені принципові помилки, неповний (невірний) розрахунок, неповна або неточна (невірна) відповідь на теоретичні питання.</li> </ul> <b>Звіт</b> з роботи не здана і незахищена без поважної причини.                                                                                       | - 0 балів   |

**Штрафні бали за:**

- 1) несвоєчасне (пізніше ніж на тиждень) подання звіту з практичного заняття (за кожен звіт, але не більше 8 балів) - 1 бал

**2.2. Модульна контрольна робота**

Модульна робота складається з 30 тестових питань. Кожне питання тесту оцінюється у 0,5 бали за такими критеріями

«Правильна відповідь» - 0,5 бали

«Не правильна відповідь» - 0 балів

Модульна контрольна робота складається з 30 тестових питань та оцінюється у 15 балів за такими критеріями

|                                                      |               |
|------------------------------------------------------|---------------|
| «Відмінно», (не менше 90% потрібної інформації)      | - 15-14 балів |
| «Добре», (не менше 75% потрібної інформації)         | - 13-11 балів |
| «Задовільно», (не менше 60% потрібної інформації)    | - 10-9 балів  |
| «Незадовільно», (не відповідає вимогам «Задовільно») | - 0 балів     |

**2.3. Домашня контрольна робота (ДКР)**

ДКР складається з 5 складових

- 1) Функціональність програми за індивідуальним завданням студента.
  - 2) Алгоритм розв'язку задачі у відповідності до вибраного методу розв'язку задачі опис послідовності дій за допомогою блок-схеми.
  - 3) Розрахунок часової складності алгоритму.
  - 4) Захист ДКР передбачає оцінку розуміння студентом програмної частини розрахункової роботи, наскільки була самостійною робота студента й наскільки він розумів поставлену задачу.
  - 5) Оформлення зовнішньої документації програми за індивідуальним завданням студента.
- Оформлення словесного алгоритму і висновків пояснювальної записки згідно загальних вимог до науково-технічних документів

#### Основні вимоги до складових ДКР

Зміст завдання полягає в створенні кожним студентом процедур на мові програмування Python, що містять алгоритми сортування, злиття та пошуку, алгоритми роботи з графами та деревами та рекурсивні алгоритми. Крім коду програми кожному студенту треба надати пояснювальну записку з розрахунком часової складності та блок-схему, а також результати тестування для різних вхідних параметрів

Кожна складова ДКР та оцінюється у 3 бали за такими критеріями

- Перераховані основні вимоги до складових ДКР повністю виконано (не менше 90%) – 3 бали;
- основні вимоги до складових ДКР виконано з зауваженнями (не менше 75%) – 2,5 бали;
- основні вимоги до складових ДКР не всі виконано (не менше 60%) – 2 бали;
- основні вимоги до складових ДКР не виконано – 0 балів.

Штрафні бали за:

- несвоєчасне (пізніше ніж на тиждень) подання звіту з практичного заняття (за кожен звіт) - 1 бал
- За кожний тиждень запізнення з поданням ДКР на перевірку нараховується штрафний –1 бал (усього не більше –5 балів)

Заохочувальні бали за:

- 1) участь у факультетській олімпіаді з дисципліни ..... 6 балів;
- 2) модернізації практичних робіт..... 6 балів;
- 3) виконання завдань із удосконалення дидактичних матеріалів з дисципліни... 6 балів;

#### **2.4. Екзаменаційна робота (Виходячи з розміру шкали $R_E = 40$ балів)**

На екзамені студенти виконують письмову контрольну роботу. Кожне завдання містить одне теоретичних запитання (завдання) та два практичних завдань. Перелік запитань рекомендований до засвоєння кредитного модуля наведений у додатку «**Діагностики результатів навчання**» до програми з навчальної дисципліни «**Основи програмування**».

Теоретичного питання – 10 балів,

Практичного завдання – 15 балів.

Максимальна кількість балів за екзаменаційну роботу дорівнює  $10 + 15 \times 2 = 40$  балів.

##### 2.4.1 Теоретичне питання оцінюється у 10 балів за такими критеріями

- повна вичерпна відповідь на питання, інформація з додаткових джерел, вдало наведені приклади – 10-9 балів;

- відповідь на питання вірна, повна, без додаткових джерел поза тем лекцій – 8-7 балів;
- відповідь на питання вірна, але не повна – 6 балів;
- відповідь на питання незадовільна – 0 балів.

#### 2.4.2 Практичне питання оцінюється у 15 балів за такими критеріями

- вичерпне, вірне і повне розв'язання задачі – 15-14 балів
- розв'язання задачі вірне, але не зовсім повне – 13-11 балів
- розв'язання задачі вірне, але неповне – 10-9 балів
- розв'язання задачі невірне – 0 балів.

#### Заохочувальні бали за:

- 1) участь у факультетській олімпіаді з дисципліни ..... 6 балів;
- 2) модернізації практичних робіт..... 6 балів;
- 3) виконання завдань із удосконалення дидактичних матеріалів з дисципліни – 6 балів

### 3. Умови позитивної проміжної атестації

Для отримання „зараховано” з першої проміжної атестації (8 тиждень) студент повинен набрати не менше, ніж 8 балів (за умови, якщо на початок 8 тижня згідно з календарним планом контрольних заходів „Ідеальний” студент має отримати 16 балів).

Для отримання „зараховано” з другої проміжної атестації (14 тиждень) студент повинен набрати не менше, ніж 15 балів (за умови, якщо на початок 14 тижня згідно з календарним планом контрольних заходів „Ідеальний” студент має отримати 30 балів).

4. **Необхідною умовою допуску до екзамену** є зарахування ДКР, всіх комп'ютерних практикумів та необхідних звітів з них, виконання МКР (не менше ніж задовільно) а також стартовий рейтинг (rC) не менше 50 % від RC, тобто 30 балів.

5. Сума рейтингових балів, отриманих студентом протягом семестру та на екзамені переводиться до підсумкової оцінки згідно з таблицею (п.6).

6. Таблиця переведення рейтингових балів до оцінок

| <b>RD = rC + rE</b>                                                                                                                               | <b>Традиційна оцінка</b> |
|---------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------|
| 95....100                                                                                                                                         | відмінно                 |
| 85....94                                                                                                                                          | Дуже добре               |
| 75...84                                                                                                                                           | добре                    |
| 65...74                                                                                                                                           | задовільно               |
| 60....64                                                                                                                                          | достатньо                |
| <b>RD ≤ 60</b>                                                                                                                                    | незадовільно             |
| <b>rC &lt; 30</b> або не виконані інші умови допуску до екзамену (не виконана ДКР, не захищені звіти з комп'ютерних практикумів, не виконана МКР) | не допущений             |

## РЕКОМЕНДОВАНИЙ СПИСОК ТЕМ ДОМАШНЬОЇ КОНТРОЛЬНОЇ РОБОТИ (ДКР)

### 1) Гра «Сапер»

#### *Правила гри:*

Прямокутне ігрове поле поділене на суміжні комірки (квадрати), деякі з яких «заміновані»; кількість «замінованих» комірок відомо. Гравець відкриває комірки, намагаючись не відкрити комірку з міною. Відкривши комірку з міною, він програє. Метою гри є відкриття всіх комірок, які не містять міни.

#### *Вимоги до програми:*

Міни повинні проставлятися після першого ходу. Якщо під відкритою коміркою міни немає, то в ній повинно з'явитися число, що показує, скільки комірок, що є сусідами щойно відкритої, «заміновано». Якщо під сусідніми комірками теж немає мін, то повинна відкритися деяка «не замінована» область до комірок, в яких є цифри.

Гравець повинен мати можливість позначати «заміновані» комірки (щоб випадково не відкрити їх) двома кольорами або типами значків: «ймовірно є міна», і «міна є точно» (в класичному «Сапері» це знак питання і прапорець, відповідно). Гравцю необхідно показувати загальну кількість мін і кількість знайдених мін.

Гра повинна мати 3 рівні складності: «Новачок» (поле 9x9, кількість мін - 10), «Любитель» (поле 16x16, кількість мін - 40) і «Професіонал» (поле 16x30, кількість мін - 90).

Забезпечити можливість зберегти гру і продовжити збережену гру. Для того, щоб гравець не міг дізнатися правильне розташування мін, бажано зберігати ігрове поле в двійковому форматі в бінарному файлі.

Реалізувати збереження результатів. Для цього необхідно реалізувати аутентифікацію гравця (запит/перевірку імені користувача і пароля перед початком гри). Для кожного гравця зберігати до 10 результатів ігор. Забезпечити можливість перегляду результатів окремого гравця і загального рейтингу гравців.

### 2) Багатофункціональний калькулятор

Написати програму (консольний або віконний додаток), що імітує роботу калькулятора з цілими і дійсними числами і виразами з дужками. Для аналізу введеної користувачем рядка з виразом не використовувати функцію `eval()`, реалізувати аналіз рядка самостійно!

Необхідно реалізувати такі операції: основні арифметичні дії (додавання, віднімання, множення, ділення), зведення в цілу ступінь, обчислення кореня, відсотків, обчислення логарифмів, факторіала, тригонометричні функції (робота з кутами в градусній і радіанній заходи, а також в системі градуси -хвилини-секунди).

Пріоритет операцій змінюють круглі дужки, вкладеність яких може бути довільною. Необхідно перевіряти правильність розстановки дужок (кількість відкриваються і закриваються дужок повинна бути однаково).

Калькулятор повинен дозволяти коригувати останнє введене число.

Створити вбудований список фізичних і математичних констант, реалізувати можливість обчислень з ними, а також можливість додавання нових констант.

Реалізувати можливість зберігання історії обчислень зі збереженням у файл за запитом користувача.

Забезпечити можливість роботи з осередком пам'яті, тобто зберігання результату і обчислення з його використанням, тобто необхідно реалізувати функції:

MS (Memory Save) - зберегти число, відображене в даний момент на дисплеї калькулятора або результат останнього обчислення в пам'ять

MR (Memory Read) - вважати число з комірки пам'яті і вивести його на дисплей

MC (Memory Clear) - очистити комірку пам'яті

M+ - додати до числа з пам'яті число, відображене на дисплеї, результат записати в пам'ять замість попереднього

M- - відняти від числа в пам'яті число, відображене на дисплеї калькулятора, результат записати в пам'ять замість попереднього.

Реалізувати можливість статистичних розрахунків. Необхідно реалізувати обчислення наступних статистичних функцій: сума, середнє арифметичне, середньоквадратичне відхилення. Забезпечити можливість ввести дані для розрахунків з клавіатури (з панелі інструментів) або зчитати дані з файлу, за вибором користувача.

Програма повинна працювати безперервно і завершувати роботу за дією користувача (натиснути відповідну кнопку на віконному додатку або набрати відповідну команду в командному додатку).

У разі помилок програма повинна видавати повідомлення і надавати можливість ввести вираз або значення знову. Приклади таких помилок:

неправильне введення користувача

невірне розташування дужок

введення операндів, при яких операції стають невірними (наприклад, розподіл на нуль, обчислення кореня з негативного числа і т.п.)

### 3) Калькулятор калорій

Написати програму (віконний або консольний додаток), що дозволяє розраховувати норму щоденного споживання калорій, білків, жирів і вуглеводів (КБЖУ) в залежності від росту, ваги, віку і способу життя людини, а також в залежності від цілей (схуднення, підтримання або набір ваги), і допомагати контролювати харчування.

*Порядок роботи програми:*

На початку роботи з користувачем програма повинна зробити запит для аутентифікації (ім'я та пароль), а також базові параметри людини (вага в кг, зріст в см, вік, стать, рівень фізичної активності).

Грунтуючись на параметрах, введених користувачем, програма повинна визначити індекс маси тіла (ІМТ), дати рекомендації щодо цілей харчування (схуднення, підтримання або набір ваги). Після чого програма з урахуванням цілей харчування повинна розрахувати добову норму споживаних калорій, а також баланс білків, жирів і вуглеводів (БЖУ).

Програма повинна давати можливість користувачеві контролювати кількість спожитих калорій, білків, жирів і вуглеводів, а також інформувати користувача про дефіцит/профіцит КБЖУ, щоб користувач міг планувати свої залишилися прийоми їжі.

Для обліку спожитих КБЖУ необхідно створити таблицю живильної (БЖУ) і енергетичної (калорії) цінності окремих продуктів і страв на 100 г. За допомогою цієї таблиці користувач повинен мати можливість підраховувати кількість спожитих КБЖУ з урахуванням ваги продукту або страви. Необхідно забезпечити можливість користувачеві додавати продукти/страви і інформацію про їх цінність в таблицю, а також розраховувати цінність страв, виходячи з їх складу.

Теоретичні відомості. Розрахунок ІМТ і КБЖУ

Формула для розрахунку індексу маси тіла (ІМТ):

$ІМТ = \text{вага, кг} / (\text{зростання, м})^2$

Таблиця 1 - Категорії ІМТ:

ІМТ < 18.5:

Нижче нормальної ваги

ІМТ > = 18.5 I < 25:

Нормальна вага

ІМТ > = 25 I < 30:

Надмірна вага

ІМТ > = 30 I < 35:

Ожиріння I ступеня

ІМТ > = 35 I < 40:

Ожиріння II ступеня



ІМТ >= 40:

Ожиріння III ступеня

Для того, щоб розрахувати добову норму КБЖУ, в першу чергу необхідно визначити базальну кількість енергії (BMR), наприклад, за формулою Харріса-Бенедикта:

для чоловіків:  $BMR = 88,362 + (13,397 * \text{вага, кг}) + (4,799 * \text{ріст, см}) + (5,677 * \text{вік})$

для жінок:  $BMR = 447,593 + (9,247 * \text{вага, кг}) + (3,097 * \text{ріст, см}) + (4,33 * \text{вік})$

BMR не враховує фізичну активність. Це потреби на метаболізм, тобто та кількість калорій, яку витрачає організм для роботи мозку, перетравлення їжі та процесів обміну речовин.

!!!Увага! Найважливіше для тих, хто бажає схуднути - менше калорій, ніж потрібно для базового обміну речовин, вживати не можна. Так людина не худне, а завдає непоправної шкоди здоров'ю !!!

Від 20% до 90% добових калорій людина витрачає на повсякденну активність. Залежно від рівня фізичної активності користувача, необхідно вибирати коефіцієнт фізичної активності (нижче наведений один з варіантів категорій фізичної активності і відповідні їм індекси):

Низька активність (сидячий спосіб життя) - 1,2;

Помірна активність (сидяча робота плюс легкі фізичні навантаження або заняття 1-3 разів на тиждень; домогосподарки) - 1,375;

Середня активність (сидяча робота плюс заняття 3-5 разів на тиждень; домогосподарка з наявністю городу; робота на ногах до 12 годин без тяжкості) - 1,55;

Висока активність (інтенсивні навантаження, заняття 6-7 разів на тиждень; робота на ногах плюс інтенсивний спорт) - 1,725;

Дуже висока активність (спортсмени і люди, що виконують схожі навантаження 6-7 разів на тиждень) - 1,9.

Для розрахунку середньої добової норми калорій, необхідно помножити BMR на коефіцієнт фізичної активності.

Коригування добової норми калорій в залежності від цілей:

Якщо мета скинути вагу, то добова норма становить 80-90% від середньої норми.

Якщо мета зберегти вагу і підтримати тонус -100%.

Якщо мета набрати масу, то добова норма становитиме 110-120% від середньої норми.

Білки, жири і вуглеводи розраховуються виходячи з необхідної суми калорій. В класичній дієтології частка білків в харчуванні має становити 10-15%, частка жирів - 25-30%, частка вуглеводів - 60-75%. При схудненні слід вибирати нижню межу по вуглеводах і жирах, а при наборі м'язової маси - трохи збільшувати білки і вуглеводи.