

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ
«ДНІПРОВСЬКА ПОЛІТЕХНІКА»



ФАКУЛЬТЕТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ
Кафедра системного аналізу та управління

Т.В. Хом'як, Т.А. Желдак, Д.М. Гаранжа

ПРАКТИКА НАВЧАЛЬНА КОМП'ЮТЕРНА

Методичні рекомендації
для здобувачів ступеня бакалавра
освітньо-професійної програми «Системний аналіз»
зі спеціальності 124 Системний аналіз

Дніпро
НТУ «ДП»
2024

Практика навчальна комп'ютерна [Електронний ресурс] : методичні рекомендації для здобувачів ступеня бакалавра освітньо-професійної програми «Системний аналіз» спеціальності 124 Системний аналіз / уклад.: Т.В. Хом'як, Т.А. Желдак, Д.М. Гаранжа ; М-во освіти і науки України, Нац. техн. ун-т «Дніпровська політехніка». – Дніпро : НТУ «ДП», 2024. – 43 с.

Укладачі:

Т.В. Хом'як, канд. ф.-м. наук, доц.;

Т.А. Желдак, канд. техн. наук, доц.;

Д.М. Гаранжа, ст. викл.

Затверджено до видання науково-методичною комісією зі спеціальності 124 Системний аналіз (протокол № 10 від 30.08.2024 року) за поданням кафедри системного аналізу та управління (протокол № 10 від 30.08.2024 р.).

Наведені мета, завдання та зміст практики навчальної комп'ютерної, а також основні вимоги до структури і порядку підготовки звіту.

Для науково-педагогічних працівників та здобувачів першого (бакалаврського) рівня вищої освіти за спеціальністю 124 Системний аналіз галузі знань 12 Інформаційні технології Національного технічного університету «Дніпровська політехніка».

Відповідальний за випуск завідувач кафедри системного аналізу та управління Т.А. Желдак, канд. техн. наук, доц.

ЗМІСТ

ВСТУП	4
1. МЕТА ТА ЗАВДАННЯ ПРАКТИКИ НАВЧАЛЬНОЇ КОМП'ЮТЕРНОЇ	4
2. ЗАВДАННЯ І ЗАГАЛЬНІ ВИМОГИ ДО ЙОГО ВИКОНАННЯ	6
2.1 Задача 1. Підсумовування функціональних рядів	6
2.2 Задача 2. Операції з масивами	11
2.3 Задача 3. Сортування масивів	18
2.4 Задача 4. Операції з файлами.	24
2.5 Задача 5. Табулювання функцій	29
2.6 Задача 6. Класи в Python	31
<i>Завдання 1.</i>	31
<i>Тести до завдання 1</i>	32
<i>Завдання 2.</i>	32
<i>Тести до завдання 2</i>	32
<i>Завдання 3.</i>	33
<i>Тести до завдання 3</i>	33
<i>Завдання 4.</i>	33
3. ВИМОГИ ДО СТРУКТУРИ, ЗМІСТУ ТА ОФОРМЛЕННЯ ЗВІТУ	35
4. КРИТЕРІЇ ОЦІНЮВАННЯ	37
РЕКОМЕНДОВАНА ЛІТЕРАТУРА	41
ДОДАТОК 1 Зразок оформлення титульної сторінки	42

ВСТУП

Практика навчальна комп'ютерна є обов'язковим елементом навчального процесу і завершальним етапом першого року підготовки здобувачів спеціальності 124 Системний аналіз. Вона має на меті узагальнення й удосконалення здобутих ними знань, практичних умінь і навичок, отриманих під час навчання.

Тривалість практики визначається затвердженою ОП та навчальним планом за спеціальністю 124 Системний аналіз. Практика здійснюється протягом чотирьох тижнів після завершення четвертої чверті. Конкретні терміни проведення практик встановлюються графіком навчального процесу, який затверджується щорічно Вченою радою НТУ «Дніпровська політехніка».

Базою проведення практики навчальної комп'ютерної є кафедра системного аналізу та управління.

Програма практики складена у відповідності з такими документами:

- Положення про проведення практики здобувачів вищої освіти Національного технічного університету «Дніпровська політехніка», що затверджене Вченою радою 11 грудня 2018 року, протокол № 15.

- Освітньо-професійна програма бакалавра зі спеціальності 124 Системний аналіз. Затверджена Вченою радою НТУ «Дніпровська політехніка» від 27.06.2024 р. протокол № 8.

- Робочі програми дисциплін підготовки здобувачів ступеня бакалавра освітньо-професійної програми «Системний аналіз» зі спеціальності 124 Системний аналіз.

1. МЕТА ТА ЗАВДАННЯ ПРАКТИКИ НАВЧАЛЬНОЇ КОМП'ЮТЕРНОЇ

Метою практики навчальної комп'ютерної є узагальнення й удосконалення здобутих теоретичних знань та практичних умінь і навичок зі спеціальності, набуття професійного досвіду та підготовка до самостійної професійної діяльності.

Відповідно до ОП «Системний аналіз» першого (бакалаврського) рівня освіти практика навчальна комп'ютерна має забезпечити такі компетентності:

- Здатність до абстрактного мислення, аналізу та синтезу;
- Здатність застосовувати знання у практичних ситуаціях;
- Здатність планувати і управляти часом;
- Знання та розуміння предметної області та розуміння професійної діяльності;
- Здатність працювати в міжнародному контексті;
- Здатність оцінювати та забезпечувати якість виконуваних робіт;
- Здатність зберігати та приумножувати моральні, культурні, наукові цінності і досягнення суспільства на основі розуміння історії та закономірностей розвитку предметної області, її місця у загальній системі знань про природу і суспільство та розвитку суспільства, техніки і технологій,

використовувати різні види та форми рухової активності для активного відпочинку та ведення здорового способу життя.

– Здатність до комп'ютерної реалізації математичних моделей реальних систем і процесів; проектувати, застосовувати і супроводжувати програмні засоби моделювання, прийняття рішень, оптимізації, обробки інформації, статистичного та інтелектуального аналізу даних;

– Здатність використовувати сучасні інформаційні технології для комп'ютерної реалізації математичних моделей та прогнозування поведінки конкретних систем а саме: об'єктно-орієнтований підхід при проектуванні складних систем різної природи, прикладні математичні пакети, застосування баз даних і знань;

– Здатність організувати роботу з аналізу та проектування складних систем, створення відповідних інформаційних технологій та програмного забезпечення.

Основними результатами навчання, що забезпечує навчальна практика, є такі:

- володіти сучасними методами розробки програм і програмних комплексів та прийняття оптимальних рішень щодо складу програмного забезпечення, алгоритмів процедур і операцій;
- створювати ефективні алгоритми для обчислювальних задач системного аналізу та систем підтримки прийняття рішень;
- проектувати, реалізовувати, тестувати, впроваджувати, супроводжувати, експлуатувати програмні засоби роботи з даними і знаннями в комп'ютерних системах і мережах

Отже, основними задачами практики навчальної комп'ютерної є наступні:

- ефективне використання базових структур даних (списки, словники, кортежі);
- реалізація та візуалізація алгоритмів обробки інформації з використанням стандартних бібліотек;
- застосування принципів об'єктно-орієнтованого програмування для реалізації завдань;
- взаємодія з зовнішніми джерелами даних (файли) та візуалізація результатів.

Завданнями навчальної комп'ютерної практики є

- закріплення навичок отриманих під час вивчення дисципліни «Програмування та алгоритмічні мови»;
- закріплення навичок візуалізації алгоритмів у вигляді блок-схем.

2. ЗАВДАННЯ І ЗАГАЛЬНІ ВИМОГИ ДО ЙОГО ВИКОНАННЯ

В ході виконання індивідуального завдання під час практики навчальної комп'ютерної кожен студент має розробити прикладну програму з використанням елементів бібліотеки TKinter, яка при завантаженні надавала б можливість отримати/перевірити результати виконання кожної з реалізованих задач.

Оформлення прикладної програми виконується в довільному, інтуїтивно зрозумілому стилі із дотриманням усіх обов'язкових вимог.

Обов'язкові вимоги до оформлення програми:

- Передбачити наявність головної форми, через яку можна запустити кожне з виконаних завдань та довідки, у якій вказується ПШБ та група автора програми.

- Перехід до кожного окремо взятого завдання та його запуск виконується з використанням допоміжних форм, на яких за допомогою елементів форми можна вносити вхідні параметри (якщо це передбачено умовою завдання).

- При переході до кожного з завдань користувач повинен бачити умову завдання, яке виконується. Представлення формул на формах виконується у вигляді рисунків.

- Для всіх без виключення завдань передбачити роботу з виключеннями для перевірки коректності введення інформації. У разі коли користувач виконав помилкове введення даних необхідно передбачити появу інформаційної «підказки», яка б надавала можливість зрозуміти причину відмови виконання програми.

- При виведенні числових даних обов'язково виводити тільки доцільну кількість десяткових знаків після коми (0, якщо мова йде про цілочисельні дані; 1-4 десяткових знаків для дійсних чисел в залежності від необхідної точності за умовами завдання).

- В залежності від умови завдання передбачити можливість вводити вхідні дані на формі, включаючи назви файлів, з якими виконується взаємодія.

- Всі завдання мають супроводжуватись роботою з виключеннями задля підвищення відмовостійкості коду та відповідними інформаційними повідомленнями, що надають користувачу розуміння стосовно помилок при введенні даних.

2.1 Задача 1. Підсумовування функціональних рядів

1. Скласти схему алгоритму і програму обчислення:

– значення функціональної суми свого варіанта (табл.1) накопиченим підсумком для числа членів i від 1 до n доданків;

– значення суми з необхідною точністю $\varepsilon = 0,001$. Визначити також кількість елементів (членів), що при цьому додаються;

– значення функції по її формулі.

2. Вказівки для складання програми і її рішення.

– Розрахунок функціонального ряду оформити у вигляді окремої функції.

– Схему алгоритму і програму скласти з урахуванням роботи з виключеннями з метою забезпечення відмовостійкості програмного коду.

– Виконати тестування і налагодження програми, забезпечивши перевірку всіх її гілок.

– Обчислення виконати для 3-х значень аргументу: двох граничних і одного проміжного.

– Результати рішення вивести у вигляді таблиці (для кожного значення аргументу), указавши кількість членів суми, значення суми і її відхилення від значення функції.

– Побудувати таблицю залежності значення суми від числа доданків.

3. Методичні вказівки до завдання.

Обчислення сум доцільно виконати методом накопичення по формулах:

$$S_0 = 0; \quad (1.1)$$

$$S_i = S_{i-1} + a_i, i = 1, 2, \dots, n; \quad (1.2)$$

де a_i – наступний елемент послідовності, який додається до суми; S_i – значення суми i доданків. Формули (1.1), (1.2) є рекурентними. Рекурентним називається співвідношення, у якому наступні значення послідовності визначаються через значення попередніх. У найпростішому випадку залежність від одного попереднього доданку має вигляд:

$$a_k = f(k, a_{k-1}), k = 1, 2, \dots \quad (1.3)$$

При рекурентних обчисленнях варто визначати початкові значення елементів послідовності (див. формулу (1.1) при обчисленні суми).

У завданні рекомендується обчислювати за допомогою рекурентних співвідношень не тільки суму, але і значення елементів функціональних рядів, де це можливо. Значне зменшення обсягу обчислень одержують, якщо елемент послідовності містить факторіали, добутки, статечні функції.

Приклад 1. Скласти рекурентні співвідношення для обчислення члена функціонального ряду вигляду:

$$a_i = (-1)^i \frac{x^{2i+1}}{2i+1} \cdot \frac{\prod_{m=1}^i (2m-1)}{\prod_{m=1}^i (2m)}; i = 0, 1, 2, \dots \quad (1.4)$$

Записуємо $i-1$ член ряду:

$$a_{i-1} = (-1)^{i-1} \frac{x^{2(i-1)+1}}{2(i-1)+1} \cdot \frac{\prod_{m=1}^{i-1} (2m-1)}{\prod_{m=1}^{i-1} (2m)}. \quad (1.5)$$

Обчислюємо співвідношення:

$$\frac{a_i}{a_{i-1}} = \frac{(-1)^i}{(-1)^{i-1}} \cdot \frac{x^{2i+1}}{x^{2(i-1)+1}} \cdot \frac{2(i-1)+1}{2i+1} \cdot \frac{\prod_{m=1}^i (2m-1)}{\prod_{m=1}^{i-1} (2m-1)} \cdot \frac{\prod_{m=1}^{i-1} (2m)}{\prod_{m=1}^i (2m)} = -x^2 \frac{(2i-1)^2}{2i(2i-1)}. \quad (1.6)$$

Звідки

$$a_i = -x^2 \frac{(2i-1)^2}{2i(2i-1)} \cdot a_{i-1}; i = 1, 2, \dots \quad (1.7)$$

Початковим значенням служить $a_0 = x$, яке обчислюємо за формулою (1.4).

Приклад 2.

$$a_i = \frac{x^i}{i!}; i = 0, 1, 2, \dots; \quad (1.8)$$

$$a_{i-1} = \frac{x^{i-1}}{(i-1)!}; \quad (1.9)$$

$$\frac{a_i}{a_{i-1}} = \frac{x^i \cdot (i-1)!}{x^{i-1} \cdot i!} = \frac{x}{i}; \quad (1.10)$$

$$a_i = \frac{x}{i} \cdot a_{i-1}; i = 1, 2, \dots \quad (1.11)$$

$$a_0 = \frac{x^0}{0!} = 1. \quad (1.12)$$

З рекурентних співвідношень видно, що обчислення члена послідовності по формулах (1.7) і (1.11) вимагає виконання значно меншого числа обчислювальних операцій, ніж по формулах (1.4) і (1.8) відповідно.

Приклад 3. Числа *Бернуллі*. У ряді формул, що визначають загальний член функціонального ряду (табл.1), є множники B_i - числа Бернуллі.

За визначенням, n -те число Бернуллі – це коефіцієнт при $-\frac{t^{2n}}{(2n)!}$, у

розкладанні в ряд за зростаючими ступенями t функції $\frac{t}{2} \operatorname{ctg} \frac{t}{2}$:

$$\frac{t}{2} \operatorname{ctg} \frac{t}{2} = 1 - B_1 \frac{t^2}{2!} - B_2 \frac{t^4}{4!} - \dots - B_n \frac{t^{2n}}{(2n)!}. \quad (1.13)$$

Значення чисел Бернуллі можна обчислювати за формулою:

$$B_n = 2 \frac{(2n)!}{(2\pi)^{2n}} \cdot \sum_{p=1}^{\infty} \frac{1}{p^{2n}}. \quad (1.14)$$

Починаючи з n , рівного 5, числа Бернуллі зростають дуже швидко зі зростанням n .

Таблиця значень чисел Бернуллі (B_n)

n	1	2	3	4	5	6	7	8	9	10	11	12	13
B_n	$\frac{1}{6}$	$\frac{1}{30}$	$\frac{1}{42}$	$\frac{1}{30}$	$\frac{5}{66}$	$\frac{691}{2730}$	$\frac{7}{6}$	$\frac{3617}{510}$	$\frac{43867}{798}$	$\frac{174611}{330}$	$\frac{854513}{138}$	$\frac{236364001}{2730}$	$\frac{853103}{6}$

4. Форма подання результатів виконання завдання.

Для кожного значення аргументу результати розв'язання представити у вигляді:

- значення аргументу $x = \dots$;
- точного значення функції $y = \dots$;
- значення сум, обчисленого з точністю $\varepsilon = 0,001$ $S_\varepsilon = \dots$;

Кількість членів суми	Значення суми	Відхилення від точного значення
1		
2		
3		
...		

Таблиця 1. Функціональні ряди (варіанти завдань)

№	Функціональні ряди	Діапазон зміни аргументу	n	Точна функція
1	$\sum_{i=0}^{n,\infty} \frac{\ln^i 3}{i!} x^i$	[0,1;1]	10	$y = 3^x$
2	$\sum_{i=0}^{n,\infty} \frac{\cos(ix)}{i}$	$\left[\frac{\pi}{5}; \frac{9\pi}{5}\right]$	16	$y = -\ln \left 2 \sin \frac{x}{2} \right $
3	$\sum_{i=0}^{n,\infty} (-1)^i \frac{x^{2i+1}}{2i+1}$	[0,1;1]	10	$y = \sin x$
4	$\sum_{i=0}^{n,\infty} (-1)^{i+1} \frac{\sin(ix)}{i}$	$\left[\frac{\pi}{5}; \frac{4\pi}{5}\right]$	18	$y = \frac{x}{2}$
5	$\sum_{i=0}^{n,\infty} \frac{x^i}{i!}$	[1;2]	15	$y = e^x$
6	$\sum_{i=0}^{n,\infty} \frac{\cos(i\pi/4)}{i!} x^i$	[0,1;1]	13	$y = e^{x \cos \frac{\pi}{4}} \cos \left(x \cdot \sin \frac{\pi}{4} \right)$
7	$\sum_{i=0}^{n,\infty} (-1)^i \frac{x^{2i}}{(2i)!}$	[0,1;1]	10	$y = \cos x$

№	Функціональні ряди	Діапазон зміни аргументу	n	Точна функція
8	$\sum_{i=0}^{n,\infty} x^i \cdot \sin\left(i \frac{\pi}{4}\right)$	[0,1;0,8]	14	$y = \frac{x \cdot \sin \frac{\pi}{4}}{1 - 2x \cdot \cos \frac{\pi}{4} + x^2}$
9	$\sum_{i=0}^{n,\infty} \frac{x^{4i+1}}{4i+1}$	[0,1;0,8]	17	$y = \frac{1}{4} \ln \frac{x+1}{1-x} + \frac{1}{2} \operatorname{arctg}(x)$
10	$\sum_{i=0}^{n,\infty} \frac{\cos(ix)}{i!}$	[0,1;1]	20	$y = e^{\cos(x)} \cos(\sin(x))$
11	$\sum_{i=0}^{n,\infty} \frac{2i+1}{i!} x^{2i}$	[0,1;1]	10	$y = (1+2x^2) \cdot e^{x^2}$
12	$\sum_{i=0}^{n,\infty} \frac{\cos\left(i \frac{\pi}{3}\right)}{i} x^i$	[0,1;0,8]	18	$y = -\frac{1}{2} \ln\left(1 - 2x \cos \frac{\pi}{3} + x^2\right)$
13	$\sum_{i=0}^{n,\infty} \frac{1}{2i+1} \left(\frac{x-1}{x+1}\right)^{2i+1}$	[0,2;1]	10	$y = \frac{1}{2} \ln x$
14	$\sum_{i=0}^{n,\infty} (-1)^i \frac{\cos(ix)}{i^2}$	$\left[\frac{\pi}{5}; \pi\right]$	20	$y = \frac{1}{4} \left(x^2 - \frac{\pi^2}{3}\right)$
15	$\sum_{i=0}^{n,\infty} (-1)^{i+1} \frac{x^{2i+1}}{4i^2 - 1}$	[0,1;1]	15	$y = \frac{1+x^2}{2} \operatorname{arctg}(x) - \frac{x}{2}$
16	$\sum_{n=0}^{n,\infty} \frac{x^{2i+1}}{(2i+1)!}$	[0,1;1]	20	$y = \frac{e^x - e^{-x}}{2} = sh(x)$
17	$\sum_{i=0}^{n,\infty} \frac{x^{2i}}{(2i)!}$	[0,1;1]	10	$y = \frac{e^x + e^{-x}}{2} = ch(x)$
18**	$\frac{1}{x} - \sum_{i=1}^{n,\infty} \frac{2^{2i} \cdot B_i}{(2i)!} x^{2i-1}$	$\left[-\frac{\pi}{2}; \frac{\pi}{2}\right]$	12	$y = ctg(x), (0 < x < \pi)$
19**	$\frac{1}{x} + \sum_{i=1}^{n,\infty} \frac{2(2^{2i-1} - 1)B_i}{(2i)!} x^{2i-1}$	$\left[-\frac{\pi}{2}; \frac{\pi}{2}\right]$	11	$y = \operatorname{cosec}(x), (0 < x < \pi)$
20	$x + \sum_{i=1}^{n,\infty} \frac{x^{2i-1} \prod_{m=1}^i (2m-1)}{\prod_{m=1}^i (2m)(2i+1)}$	[-0,8;0,8]	14	$y = \arcsin(x), (x < 1)$

№	Функціональні ряди	Діапазон зміни аргументу	n	Точна функція
21	$x + \sum_{i=1}^{n,\infty} (-1)^i \frac{x^{2i-1} \prod_{m=1}^i (2m-1)}{(2i+1) \prod_{m=1}^i (2m)}$	$[-0,5; 0,5]$	12	$y = \operatorname{arcsinh}(x) = \ln(x + \sqrt{x^2 + 1}), (x < 1)$
22**	$\ln x - \sum_{i=1}^{n,\infty} \frac{2^{2i-1} \cdot B_i \cdot x^{2i}}{i(2i)!}$	$\left[-\frac{3\pi}{4}; \frac{3\pi}{4}\right]$	13	$y = \ln \sin x , (0 < x < \pi)$
23**	$-\sum_{i=1}^{n,\infty} \frac{2^{2i-1} (2^{2i} - 1) \cdot x^{2i-1} B_i}{(2i)!}$	$\left[-\frac{\pi}{4}; \frac{\pi}{2}\right]$	10	$y = \ln(\cos x), (x < \pi/2)$
24**	$\sum_{i=1}^{n,\infty} \frac{(-1)^{i+1} 2^{2i} \cdot x^{2i-1} B_i}{(2i)!}$	$\left[-\frac{\pi}{3}; \frac{\pi}{3}\right]$	12	$y = \frac{e^x - e^{-x}}{e^x + e^{-x}}, (x < \pi/2)$
25**	$\frac{1}{x} + \sum_{i=1}^{n,\infty} \frac{(-1)^{i+1} x^{2i-1} 2^{2i} \cdot B_i}{(2i)!}$	$\left[-\frac{3\pi}{4}; \frac{3\pi}{4}\right]$	13	$y = \frac{e^x + e^{-x}}{e^x - e^{-x}}, (0 < x < \pi)$
26**	$\frac{1}{x} + \sum_{i=1}^{n,\infty} \frac{(-1)^i 2x^{2i-1} (2^{2i-1} - 1) \cdot B_i}{(2i)!}$	$\left[-\frac{\pi}{2}; \frac{\pi}{2}\right]$	10	$y = \frac{2}{e^x - e^{-x}}, (0 < x < \pi)$
27**	$1 - \frac{x}{2} + \sum_{i=1}^{n,\infty} \frac{(-1)^{i+1} x^{2i} \cdot B_i}{(2i)!}$	$[-\pi; \pi]$	13	$y = \frac{x}{e^x - 1}, (x < 2\pi)$
28*	$\sum_{i=0}^{n,\infty} \frac{x^i \cdot \sin\left(a + \frac{i\pi}{2}\right)}{i!}$	$[-2\pi; 2\pi]$	11	$y = \sin(x + a), (x < \infty)$
29*	$\sum_{i=0}^{n,\infty} \frac{(x \ln a)^i}{i!}$	$[-5; 5]$	15	$y = a^x = e^{x \ln a}, (x < \infty)$
30	$x + \sum_{i=1}^{n,\infty} (-1)^i \frac{2^{2i} x^{2i+1}}{(2i)!}$	$[-5; 5]$	14	$y = x \cos(2x), (-\infty < x < \infty)$

* - Значення параметра a студент обирає самостійно.

** B_i – це числа Бернуллі, значення і правила обчислення яких наведені вище.

2.2 Задача 2. Операції з масивами

1. Скласти блок-схему і програму, що розв'язує поставлену задачу з використанням зазначеної структури даних – векторів і матриць.

2. Вказівки для складання програми і її виконання.

– Програма повинна містити в собі:

- 1) операції введення структур із клавіатури (чи заповнення структури за зазначеним правилом) та у разі можливості передбачити заповнення випадковими числами;
- 2) вивід структур на екран, а також деякі стандартні операції обробки вказаних структур (скалярний добуток векторів, множення матриць і множення матриці на вектор, пошук мінімального, максимального і середнього елементів структур).

– При виконанні зазначених операцій використовувати найбільш придатний вигляд операторів циклу.

– Блок-схема алгоритму повинна бути складена з урахуванням обраних операторів.

– Передбачити роботу з виключеннями з метою забезпечення відмовостійкості програмного коду.

3. Варіанти завдань.

1. Скласти схему алгоритму і програму для обчислення білінійної форми

$$B = m \sum_{i=1}^n \left(x_i \sum_{k=1}^n (a_{ik} \cdot y_k) \right),$$

де m - максимальний елемент вектора $\bar{x} = (x_1, \dots, x_n)$. Матриця $A = [a_{ij}]_{n \times n}$, вектор \bar{x} і вектор $\bar{y} = (y_1, \dots, y_n)$ задані ($n \leq 6$).

2. Скласти схему алгоритму і програму, що з вектора $\bar{x} = (x_1, \dots, x_n)$; ($n \leq 20$) виділяє вектор $\bar{y} = (y_1, \dots, y_m)$; ($m \leq n$) за правилом: елемент вектора \bar{x} є елементом вектора \bar{y} , якщо квадратне рівняння $x^2 - 2p_i x + q = 0$ має дійсні і різні корені.

3. Елементи вектора $\bar{a} = (a_1, \dots, a_{10})$ розташовані в порядку зростання за абсолютною величиною. Задані k - ціле і y - дійсне. Скласти програму для розв'язання наступної задачі: побудувати і вивести на екран вектор $\bar{b} = (b_1, \dots, b_{10})$, який містить елементи вектора \bar{a} крім k -го, а також вставлений на потрібне місце y так, щоб елементи вектора \bar{b} виявилися також розташованими в порядку зростання по абсолютній величині.

4. Дано вектор $\bar{x} = (x_1, \dots, x_n)$, $n \leq 15$. Здійснити циклічне зрушення елементів цього вектора:

а) уліво на одну позицію, тобто одержати вектор $\bar{x} = (x_2, x_3, \dots, x_n, x_1)$;

б) вправо на дві позиції, тобто одержати вектор $\bar{x} = (x_{n-1}, x_1, \dots, x_{n-2})$.

Вивести на екран вихідний вектор і обидва перетворених вектори.

5. Значення елементів вектора $\bar{x} = (x_1, \dots, x_{10})$ обчислити за формулою

$$x_i = 1,5 \lg(0,5i), i = 1,2,\dots,10.$$

Отриманий вектор вивести на екран.

Вектор \bar{x} і заданий вектор \bar{y} перетворити за правилом: більше з x_i, y_i прийняти як нове значення x_i , а менше - як нове значення y . Перетворені вектори \bar{x} і \bar{y} вивести на екран.

6. Заданий вектор $\bar{x} = (x_1, \dots, x_{10})$ дійсних чисел. Елементи вектора $\bar{y} = (y_1, \dots, y_{10})$ обчислити за формулою:

$$y_i = 0,26 \ln(1 + \sqrt{i}).$$

Логічній змінній a присвоїти значення TRUE, якщо змінна $S = (x_1 y_{10})^2 + (x_2 y_9)^2 + \dots + (x_{10} y_1)^2$ належить відрізку $[0;1]$ і значення FALSE у протилежному випадку. Вектори \bar{x} , \bar{y} , а також значення S і a вивести на екран.

7. Значення елементів векторів $\bar{x} = (x_1, \dots, x_{15})$ і $\bar{y} = (y_1, \dots, y_{15})$ обчислити за формулами:

$$x_i = \operatorname{arctg} \frac{i+2}{15}; y_i = e^{i \cos \sqrt{15}}.$$

Отримані вектори вивести на екран. З векторів \bar{x} і \bar{y} одержати вектор $\bar{z} = (x_1, y_1, x_2, y_2, \dots, x_{15}, y_{15})$, елементи якого перенумеровані від 1 до 30. Отриманий вектор вивести на екран.

8. Задано логічний $\bar{y} = (y_1, \dots, y_n), n \leq 10$ і дійсний $\bar{x} = (x_1, \dots, x_n)$ вектори. Перетворити вектор \bar{x} за правилом: якщо y_i має значення TRUE, то x_i помножити на 10. Інакше – змінити знак x_i на протилежний.

9. Задано вектор, що містить цілі числа $\bar{y} = (y_1, \dots, y_{16})$. Побудувати вектор $\bar{x} = (x_1, \dots, x_{16})$, прийнявши в якості перших його елементів усі від'ємні елементи вектора \bar{y} (зі збереженням порядку їхнього проходження), а як інші елементи всі додатні елементи вектора \bar{y} також зі збереженням порядку їхнього проходження.

10. Задано вектор, що містить цілі числа $\bar{y} = (y_1, \dots, y_n), n \leq 20$. Якщо серед елементів цього вектора є елемент зі значенням, рівним цілому числу b (задане), то змінній l присвоїти значення, рівне сумі всіх елементів, що передують цьому елементу. У протилежному випадку покласти $l = 0$.

11. Значення елементів вектора $\bar{y} = (y_1, \dots, y_{19})$ обчислити за формулою:

$$y_i = 1,2 \sin(0,1i) + 0,4 \cos(i-1), i = 1,2,\dots,19.$$

Отриманий вектор вивести на екран.

Знайти найбільший елемент вектора \bar{y} , його номер k вивести на екран. Всі елементи вектора \bar{y} з парними номерами, що передують найбільшому елементу y_k , помножити на y_k . Інші елементи залишити без зміни. Перетворений вектор вивести на екран.

12. Задано дійсний вектор $\bar{y} = (y_1, \dots, y_n), n \leq 20$. Якщо хоча б один елемент цього вектора менше, ніж $t = -2$, то всі від'ємні елементи замінити їхніми квадратами, залишивши інші елементи без зміни; у протилежному випадку всі елементи вектора \bar{y} помножити на 0,1. Вихідний та перетворений вектори вивести на екран.

13. Обчислити значення функції

$$z = \sum_{i=1}^{15} \prod_{k=1}^{10} \frac{a_i + b_k}{2},$$

де значення a_i задані масивом $(a_1, a_2, \dots, a_{15})$, а значення b_k змінюється від 1 до 1,9 із кроком 0,1.

14. Знайти і запам'ятати десять найбільших елементів масиву $(a_1, a_2, \dots, a_{50})$. Вивести на екран самі елементи і їхні номери.

15. Перемножити матриці $A = [a_{i,j}]_{n \times m}$ і $B = [b_{i,j}]_{m \times l}$. Елементи результуючої матриці обчислити за допомогою виразу

$$C_{i,k} = \sum_{j=1}^m a_{i,j} b_{j,k}.$$

Знайти найбільший і найменший елементи матриці C і номери рядків і стовпців, у яких вони розміщені. Результати вивести у вигляді матриці.

16. Переписати перші додатні елементи кожного рядка матриці $A = [a_{i,j}]_{10 \times 20}$ в масив Y . Якщо в рядку немає додатного елемента, то варто записати нуль у масив Y . З циклу можливі два виходи: 1) за допомогою умовного оператора, якщо $x_{ij} > 0$, до оператора, що записує x_{ij} в масив Y ; 2) природній - до оператора, що стоїть за циклом та записує 0 у масив Y .

17. Знайти значення аргументу x для функції $y = a \cdot e^{bx+cx^2}$, при якому досягається максимум з точністю $\varepsilon = 0,005$. Значення x змінюється від -2 до 2 із кроком 0,2. Функція має один максимум.

18. Скласти схему алгоритму і програму для зарахування на стипендію, якщо вона виплачується тим, хто за результатами сесії отримав оцінки "4" і "5"

чи має одну оцінку "3", але активно займається суспільною роботою. Студент, що має всі оцінки "5", отримує підвищену стипендію.

Відомості про результати сесії для групи зведені в матрицю $A = [a_{i,j}]_{n \times 10}$. У стовпцях 2-8 приведені оцінки, отримані на сесії, у стовпці 1 зазначене "1", якщо студент бере активну участь у суспільній роботі, чи "0" - якщо не приймає.

У стовпець 9 варто помістити середній бал студентів, а ознаку стипендії (1 – звичайна, 2 – підвищена, 0 – стипендії немає) помістити в стовпець 10.

При рішенні перевіряється наявність незадовільних оцінок. Якщо вони є, то середній бал $a_{i,9}$ студента варто вважати рівним нулю й у стовпець 10 записувати ознаку 0. Якщо незадовільних оцінок немає, то варто підраховувати суму балів для кожного студента, а після закінчення циклу - середній бал. Для призначення на стипендію необхідно підрахувати кількість відмінних n_5 і задовільних n_3 оцінок. Якщо після закінчення циклу виявиться, що $n_5 = 7$, то призначається підвищена стипендія (записується ознака 2 у стовпець 10). В іншому випадку необхідно перевірити кількість задовільних оцінок. Щоб зменшити обсяг програми, з кількості задовільних оцінок n_3 варто відняти ознаку суспільної роботи n_1 . Ця різниця буде менше чи дорівнювати нулю у випадку, якщо студент не має оцінок "3" чи має одну оцінку "3", але активно займається суспільною роботою. В обох випадках стипендія студенту призначається, тобто в стовпець 10 записується ознака 1. Якщо різниця більше нуля, то стипендія не нараховується.

19. Скласти схему алгоритму і програму, що дозволяє визначити за результатами сесії кількість студентів n_5 , що мають всі оцінки "5", студентів n_4 , що мають всі оцінки "4", студентів n_{45} , що мають всі оцінки "5" і "4", студентів n_3 , що мають хоча б одну оцінку "задовільно", студентів n_2 , що мають хоча б одну оцінку "незадовільно" чи незданий іспит. Результати сесії зведені в матрицю $A = [a_{i,j}]_{n \times 7}$. Для рішення цієї задачі необхідно підраховувати для кожного студента кількість відмінних k_5 , гарних k_4 , задовільних k_3 оцінок. Якщо виявиться, що $k_5 = 7$, то n_5 необхідно збільшити на одиницю; якщо $k_4 = 7$, то треба збільшити на одиницю n_4 ; якщо $k_5 + k_4 = 7$, то варто збільшити на одиницю n_{45} ; у протилежному випадку - на одиницю n_3 . Якщо $a_{ij} \leq 2$ (не зданий іспит – за умовчанням 0), то необхідно відразу виходити з циклу і збільшувати на одиницю n_2 .

20. Скласти схему алгоритму і програму для визначення місць, зайнятих у змаганні десятиборцями. Результати, показані 20 спортсменами, поміщені в матрицю $A = [a_{i,j}]_{20 \times 13}$. У стовпці 1 матриці вказаний номер спортсмена, у стовпцях з 2-го по 11-й зазначені бали, набрані спортсменом у різних видах

програми. У стовпці 12 варто помістити суми балів, набраних спортсменами, у стовпці 13 - номери спортсменів у порядку зайнятих ними місць у змаганні.

Розв'язання задачі зводиться до обчислення сум елементів кожного рядка матриці, починаючи з 2-го по 11-й стовпець, і запису підсумкових сум у стовпець 12. Потім необхідно елементи стовпця 12 упорядкувати в порядку убування набраних спортсменами сум, а в стовпці 13 переставити номери спортсменів відповідно до сум набраних балів. Спортсмени, що мають однакову суму балів, розташовуються в порядку зростання їхніх номерів (попередніх).

21. Скласти схему алгоритму і програму рішення завдання варіанта №20, що упорядковує всі рядки матриці A відповідно до набраної суми балів спортсменами. Результати спортсменів вводяться в будь-якій послідовності. Крім того, для тимчасового збереження після введення даних про одного спортсмена варто виділити одномірний масив, що складається з 11 елементів. Для упорядкування результатів кожного спортсмена відповідно з зайнятим їм місцем необхідно організувати цикл, у якому будуть мінятися місцями всі елементи (з 2-го по 11-й) рядків з номерами i_{\max} і k .

22. Скласти схему алгоритму і програму для обчислення і запам'ятовування в масиві $U(3)$ найбільших коренів квадратних рівнянь:

$$ax^2 + bx - 4 = 0;$$

$$y^2 - cy + d = 0;$$

$$2z^2 + dz - 1 = 0.$$

Якщо корінь комплексний, то вважати його рівним нулю. Корені рівняння шукати в процедурі. Коефіцієнти рівнянь вводяться з клавіатури.

23. Скласти схему алгоритму і програму для обчислення $z = \frac{x+y}{kn}$, де x і k - сума і кількість додатних елементів масиву $A(40)$; y и n - сума і кількість від'ємних елементів масиву $B(50)$. Для обчислення суми і кількості додатних і від'ємних елементів масиву використовувати однакову процедуру загального вигляду. Нульових елементів у масивах немає.

24. Скласти схему алгоритму і програму для обчислення функції

$$Z = \begin{cases} xa_{\max}, \text{ при } \dots x < y; \\ yb_{\min}, \text{ при } \dots x \geq y; \end{cases}$$

де $x = \sum_{i=1}^{20} a_i$ і $y = \sum_{j=1}^{15} b_j$ - суми елементів масивів; a_i і b_j - цілі числа.

В одній процедурі обчислюється сума елементів масиву, а в іншій процедурі - знайти найбільший і (чи) найменший елемент масиву.

25. Скласти схему алгоритму і програму для знаходження найбільшого загального дільника двох цілих додатних чисел шляхом обчислення:

$$k_1 = \max(n, m); k_2 = \min(n, m);$$

$$k_3 = \max(k_1 - k_2, k_2) - \min(k_1 - k_2, k_2) \dots;$$

$$k_i = \max(k_{i-2} - k_{i-1}, k_{i-1}).$$

Якщо $k_i = k_{i-2}$ то k_i - найбільший загальний дільник. Для знаходження \max і \min використовувати процедуру загального вигляду. При виконанні необхідно багаторазово звертатися до процедури знаходження найбільшого і найменшого значень змінних. Перше звертання дозволить знайти початкові значення k_1 та k_2 . Друге звертання до процедури здійснюється з ітераційного циклу багаторазово для обчислення значень k_3, k_4, \dots, k_i . Цикл варто повторювати, доки k_i не стане рівним k_{i-2} .

26. Скласти схему алгоритму і програму для обчислення середнього значення Sx і дисперсії Dx для моментів появи випадкових подій t_1, t_2, \dots, t_{100} і інтервалів між ними, використовуючи процедуру. Перед обчисленням Sx і Dx для інтервалів між випадковими подіями необхідно упорядкувати елементи масиву T по зростанню, обчислити і записати в масив $T1$ інтервали між суміжними t_i : $t1_i = t_i - t_{i-1}; t_0 = 0$. Упорядкування елементів по зростанню здійснити в основній процедурі.

27. Скласти схему алгоритму і програму для визначення місць, зайнятих гімнастами на змаганнях, використовуючи процедуру. Змагання проводилися в три потоки - юніори, першорозрядники, майстри спорту. Результати, показані гімнастами на окремих снарядах, для кожного потоку зведені в матриці $A = [a_{i,j}]_{10 \times 6}$, $B = [b_{i,j}]_{12 \times 6}$ і $C = [c_{i,j}]_{15 \times 6}$.

Для виконання цього завдання в процедурі необхідно обчислити суму балів, отриману кожним гімнастом, і упорядкувати їхні результати в порядку убутання сум. У результаті вийде матриця R , що складається з n рядків і двох стовпців, де n - кількість спортсменів у потоці. У першому стовпці матриці R - упорядкування по убутанню суми балів, а в другому - номери спортсменів.

28. Скласти схему алгоритму і програму для обчислення функції

$$Z = \frac{\sum_{i=1}^{20} |a_i| + \sum_{i=1}^{30} |b_i|}{\sum_{i=1}^{20} a_i} \dots$$

Обчислення сум виконати в процедурі.

29. Скласти схему алгоритму і програму для обчислення функції

$$z = \left(\prod_{i=1}^n e^{a_i} - \prod_{i=1}^m e^{b_i} \right) / \left(\prod_{i=1}^{n+m} c_i^2 \right)$$

де a_i, b_i, c_i - елементи масивів, що складаються відповідно з n , m і $n + m$ - елементів. Добутки обчислювати в окремій функції загального вигляду.

30. Скласти схему алгоритму і програму для обчислення функції

$$Z = \frac{(a - x_1)(a - x_2)}{(b - y_1)(b - y_2)}$$

де x_1 і x_2 - корені рівняння $ax^2 - x + 1 = 0$; y_1 і y_2 - корені рівняння $2y^2 + y - b = 0$. Якщо корені уявні, то вважати їх рівними нулю. Для обчислення коренів використовувати процедуру.

2.3 Задача 3. Сортування масивів

1. Скласти схему алгоритму і програму, що буде сортувати одновимірний масив з N елементів

2. *Методичні вказівки до завдання.* Відомості про алгоритми сортування масивів.

Сортування вибором.

Виконується $(n - 1)$ кроків порівнянь і перестановок. На i -ому кроці $(1 \leq i \leq n - 1)$ знаходиться найбільший з $j = n + 1 - i$ елементів масиву і переставляється з j -м елементом. Так, на першому кроці $(i = 1)$ знаходиться найбільший з $j = n + 1 - 1 = n$ елементів, який і переставляється на j -у позицію, а на останньому кроці $(j = n - 1)$ знаходиться найбільший з $j = n + 1 - (n - 1) = 2$ елементів і переставляється з другим елементом. У результаті елементи масиву розмістяться в порядку зростання.

Сортування обмінами.

Послідовним переглядом чисел a_1, \dots, a_n знайти найменше i таке, що $a_i > a_{i+1}$. Поміняти a_i і a_{i+1} місцями, відновити перегляд з елемента a_{i+1} і т.д. Тим самим найбільше число пересунеться на останнє місце. Наступні перегляди починати знову спочатку, зменшуючи на одиницю кількість елементів, що переглядаються. Масив буде упорядкований після перегляду, у якому брали участь тільки перший і другий елементи.

Сортування простими вставками.

Переглядати послідовно a_2, \dots, a_n і кожен новий елемент a_i ставити на придатне місце у вже впорядковану сукупність a_1, \dots, a_{i-1} . Це місце визначається послідовним порівнянням a_i з упорядкованими елементами a_1, \dots, a_{i-1} .

Сортування методом бульбашки.

Робимо послідовний перегляд масиву, і щораз (пару за парою) порівнюємо сусідні числа. Якщо числа в парі розташовані в порядку зростання, залишаємо їх без зміни; в іншому випадку змінюємо їх місцями. Потім (у будь-якому випадку) переходимо до наступної пари. Сортування вважається закінченим, якщо у ході перегляду не була зроблена жодна перестановка.

Приклад. Розглянемо масив із шести чисел.

Вихідний масив	Після першого перегляду	Після другого перегляду	Після третього перегляду	Після четвертого перегляду сортування закінчене
6	7	7	7	7
2	6	6	6	6
4	2	5	5	5
5	4	2	4	4
7	5	4	2	2
1	1	1	1	1

Алгоритм методу бульбашки.

1. Нехай k – індекс (номер) самого верхнього елемента, порівнюваного в даному перегляді. При першому перегляді $k = n$.

2. T – ознака виконання перестановки при перегляді. Вважаємо $T = 0$.

3. При кожному значенні $j = 1, 2, \dots, k - 1$ виконуємо порівняння a_j із a_{j+1} .

Якщо $a_j > a_{j+1}$ змінюємо їх місцями і приймаємо $T = j$.

4. Якщо обмінів при перегляді не було ($T = 0$), сортування виконане. У противному випадку встановлюємо $k = T$ та повертаємося до пункту 2.

Сортування методом Шелла.

Спочатку окремо групуються і сортуються елементи, що знаходяться один від одного на відстані 4. Такий процес називається четвертним сортуванням. Після першого проходу елементи перегруповуються - тепер кожен елемент групи стоїть від іншого через дві позиції - і знову сортуються. Це називається подвійним сортуванням. І нарешті, на третьому проході йде звичайне чи одинарне сортування.

На перший погляд можна засумніватися: якщо необхідно кілька процесів сортування, причому в кожний включаються всі елементи, то чи не додадуть вони більше роботи, чим заощадять часу? Однак на кожному етапі або сортується відносно мало елементів, або елементи вже досить добре упорядковані і потрібно порівняно небагато перестановок.

Ясно, що такий метод у результаті дає упорядкований масив, і, звичайно ж, відразу видно, що кожен прохід від попередніх тільки виграє (тому що кожне i -те сортування поєднує дві групи, уже відсортовані $2i$ -тим сортуванням). Так само очевидно, що відстані в групах можна зменшувати по-різному, аби останнє було одиничним, адже в найгіршому випадку останній прохід і зробить

усю роботу. Усі t відстані позначаються відповідно H_1, H_2, \dots, H_n . Для них виконуються умови

$$H_t = 1, H_{i+1} < H_i;$$

Кожне H -сортування програмується як сортування за допомогою прямого включення. Причому простота умови закінчення пошуку місця для включення забезпечується методом бар'єрів. Ясно, що кожне із сортувань має потребу у встановленні свого власного бар'єра, і програму для визначення його місця розташування необхідно робити наскільки можливо простою. Тому доводиться розширювати масив не тільки на один-єдиний компонент a_0 , а на H_1 компонентів.

Аналіз цього алгоритму поставив кілька проблем. Не відомо, які відстані дають найкращі результати. Д.Батіг у роботі "Мистецтво програмування для ЕОМ" показує, що має сенс використовувати таку послідовність (записана в зворотному порядку):

$$1, 4, 13, 40, 121, \dots \text{чи } 1, 3, 7, 15, 31, \dots$$

Математичний аналіз показує, що в останньому випадку для сортування n елементів методом Шелла витрати пропорційні $n^{1/2}$

Вставка в упорядкований масив.

Нехай заданий упорядкований за зростанням масив цілих або дійсних чисел $a_1 \leq a_2 \leq \dots \leq a_n$ та деяке число b (відповідно ціле чи дійсне), для якого потрібно знайти таке місце серед чисел a_1, \dots, a_n , щоб після вставки числа b на це місце упорядкованість не порушилася. Якщо внаслідок рівності між собою деяких елементів масиву число b можна вставляти на різні місця, то потрібно визначити найближче до початку масиву місце. Ця задача називається задачею пошуку місця елемента. Для $b \in n+1$ можливості розташування: $b \leq a_1, a_1 < b \leq a_2, \dots, a_{n-1} < b \leq a_n, a_n < b$ і розв'язком задачі пошуку місця елемента b буде одне з чисел $1, \dots, n+1$. Для розв'язання задачі доцільно використовувати алгоритм бінарної вставки.

Спочатку границі пошуку місця елемента дорівнюватимуть 1 та $n+1$. В подальшому ці границі змінюються в такий спосіб: елемент b порівнюється з a_s , де S – ціла частина середнього арифметичного границь; якщо $a_s < b$, то замінити колишню нижню границю на $S+1$, а верхню залишити без зміни, інакше, залишити без зміни нижню границю, а верхню замінити на S . Цей процес закінчується, коли обидві границі дорівнюватимуть деякому числу t – місцю для вставки нового елемента в упорядкований масив. Число порівнянь, що виконує цей алгоритм, не перевищує $[\log_2(n+1)] + 1$.

Злиття двох упорядкованих масивів (метод Дж. фон Неймана).

Метод заснований на злитті двох упорядкованих масивів в один, упорядкований за тією ж ознакою. При цьому спочатку аналізуються перші записи обох масивів. Той запис, у якого значення ключа менше, переписується в новий масив. Запис, що залишився, послідовно порівнюється з записами з

іншого масиву. У новий масив після кожного порівняння попадає запис з меншим значенням ключа. Запис, що залишився, порівнюється із записами з іншого масиву. Процес продовжується до вичерпання записів одного з масивів. Потім залишок іншого масиву дописується в новий масив.

В задачах, що розглядаються, взято припущення, що масиви А і В містять N і M упорядкованих у зростаючому порядку дійсних чисел. Потрібно об'єднати ці масиви так, щоб отримати масив С, у якому елементи розташовуються в зростаючому порядку. Масив С повинний містити N+M елементів.

3. Варіанти завдань.

1. Скласти схему алгоритму і програму упорядкування (сортування) масиву a_1, a_2, \dots, a_n за зростанням ($a_1 \leq a_2 \leq \dots \leq a_n$) методом вибору.

2. Те ж – методом обмінів.

3. Те ж – методом простих вставок.

4. Те ж – методом бульбашки.

5. Те ж – методом Шелла.

6. Дана дійсна прямокутна матриця $A = [a_{i,j}]_{m \times n}$. Скласти програму, що складається з головного модуля і процедури перестановки рядків матриці за зростанням перших елементів рядків.

7. Те ж – за зростанням сум елементів рядків.

8. Те ж – за неспаданням добутків елементів стовпців.

9. Те ж – за незростанням абсолютних значень найменших елементів рядків.

10. Те ж – за неспаданням кількості додатних елементів стовпців.

11. Те ж – за незростанням середніх значень елементів рядків.

12. Заданий масив дійсних чисел $\bar{a} = [a_i]_n$. Скласти програму перестановки елемента масиву $(a_1; a_2; \dots; a_n)$ таким чином, щоб на початку масиву йшла група елементів, більших елемента, що в початковому масиві розташовувався на першому місці, потім – сам елемент, потім – група елементів, що менші або дорівнюють йому. Число порівнянь або переміщень не повинно перевищувати $n - 1$.

13. Заданий упорядкований за неспаданням масив дійсних чисел $\bar{a} = [a_i]_n$, та число b . Скласти програму вставки числа b в масив \bar{a} таким чином, щоб упорядкованість не порушувалася.

14. Те ж, але b_i – елементи масиву $\bar{b} = [b_j]_m$.

15. Те ж, але попередньо видалити з масиву $\bar{a} = [a_i]_n$, k -ий елемент ($1 \leq k \leq n$), а потім уставити число b таким чином, щоб упорядкованість не порушувалася.

16. Є упорядковані за неспаданням масиви дійсних чисел $\bar{a} = [a_i]_n$ та $\bar{b} = [b_j]_m$. Скласти програму злиття масивів \bar{a} і \bar{b} в упорядкований за неспаданням масив $\bar{c} = [c_i]_{n+m}$.

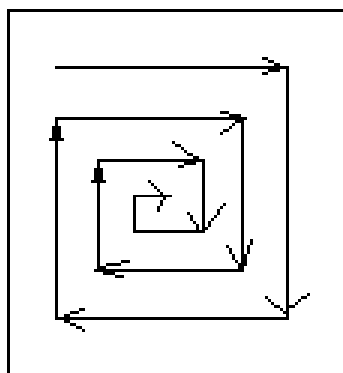


Рис. 3.2

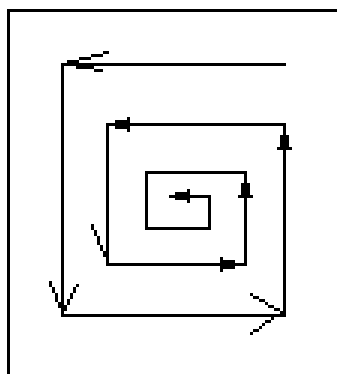


Рис. 3.3

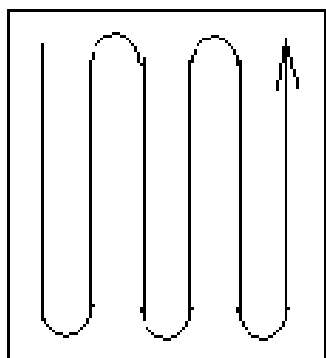


Рис. 3.4

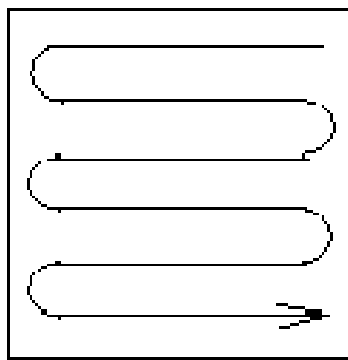


Рис. 3.5

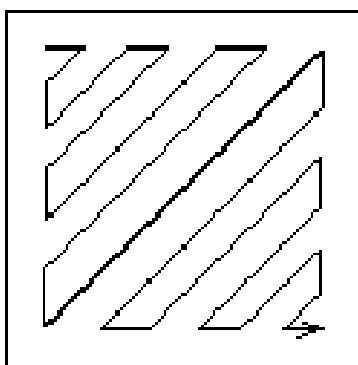


Рис. 3.6

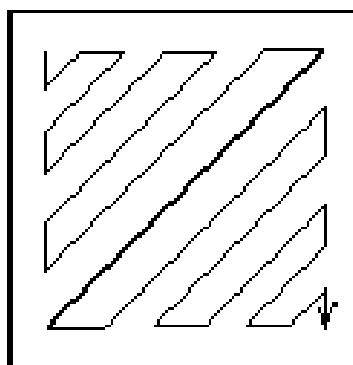


Рис. 3.7

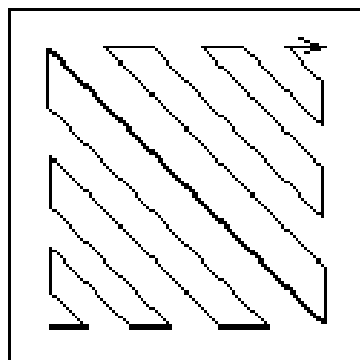


Рис. 3.8

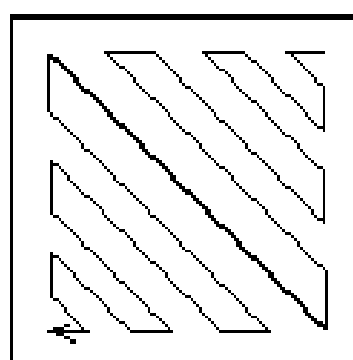


Рис. 3.9

17. Задана квадратна матриця $A = [a_{i,j}]_{m \times n}$ дійсних чисел. Програму розміщення елементів матриці A по незростанню в напрямку, зазначеному на рис 3.2.

- 18. Те ж за рис. 3.3.
- 19. Те ж за рис. 3.4.
- 20. Те ж за рис. 3.5.
- 21. Те ж за рис. 3.6.
- 22. Те ж за рис. 3.7.
- 23. Те ж за рис. 3.8.
- 24. Те ж за рис. 3.9.

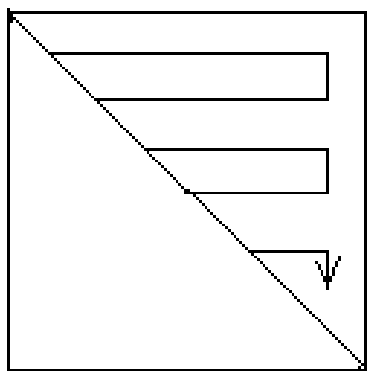


Рис. 3.10

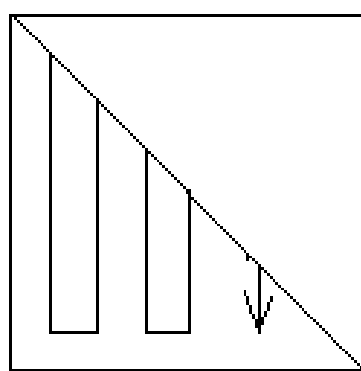


Рис. 3.11

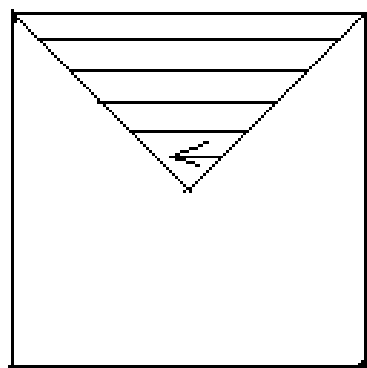


Рис. 3.12

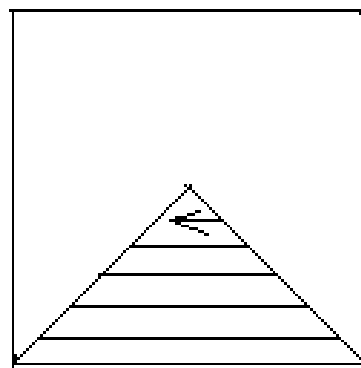


Рис. 3.13

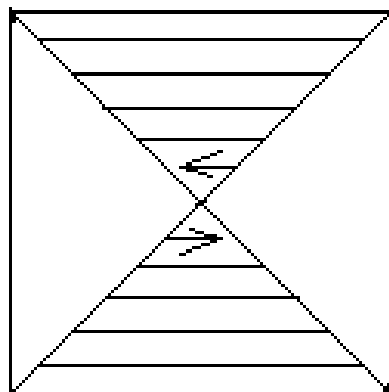


Рис. 3.14

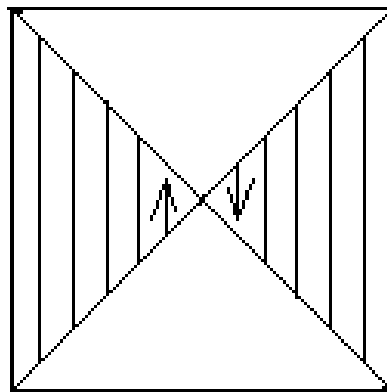


Рис. 3.15

25. Дана квадратна матриця цілих чисел $A = [a_{i,j}]_{m \times n}$. Скласти програму формування вектора \bar{b} , що містить непарні числа, розташовані у заштрихованій частині матриці (рис. 3.10). Розмістити елементи вектора \bar{b} в порядку зростання.

26. Те ж за рис. 3.11.

27. Те ж за рис. 3.12.

28. Те ж за рис. 3.13.

29. Те ж за рис. 3.14.

30. Те ж за рис. 3.15.

2.4 Задача 4. Операції з файлами.

1. Скласти схему алгоритму і програму обробки текстової інформації:

- Прочитати інформацію з цього файлу.
- Виконати необхідні дії з отриманими даними і представити результат у необхідному вигляді.

2. Вказівки для складання програми і її рішення.

- Схему алгоритму і програму скласти з перевіркою існування файлу на диску.
- Передбачити можливість роботи з файлами невизначеної заздалегідь довжини.
- Виконати тестування і налагодження програми, забезпечивши перевірку всіх її гілок.

3. Варіанти завдань.

1. Дано файл f , компоненти якого є дійсними числами. Знайти:

- а) суму компонентів файлу;
- б) добуток компонентів файлу;
- в) суму квадратів компонентів файлу;
- г) модуль суми і квадрат добутку компонент файлу;
- д) останній компонент файлу.

2. Дано файл f , компоненти якого є дійсними числами. Знайти:

- а) найбільше зі значень компонентів;
- б) найменше зі значень компонентів з парними номерами;
- в) найбільше зі значень модулів компонент із непарними номерами;
- г) суму найбільшого і найменшого зі значень компонентів;
- д) різницю першого й останнього компонентів файлу.

3. Дано файл f , компоненти якого є цілими числами. Знайти:

- а) кількість парних чисел серед компонентів;

- б) кількість подвоєних непарних чисел серед компонентів;
- в) кількість квадратів непарних чисел серед компонентів.

4. Дано символні файли f і g . Переписати із збереженням порядку проходження компоненти файлу f в g , а компоненти файлу g - у файл f . Використовувати допоміжний файл h .

5. Дано символний файл f . У файлі f не менше двох компонентів. Визначити, чи є два перших символи файлу цифрами. Якщо так, то встановити, чи є число, утворене цими цифрами, парним.

6. Дано файл f , компоненти якого є цілими числами. Одержати у файлі g усі компоненти файлу f

- а) які є парними числами;
- б) діляться на 3 і не діляться на 7;
- в) які є точними квадратами.

7. Дано файл f , що містить різні дати. Кожна дата – це число, місяць і рік. Знайти:

- а) рік з найменшим номером;
- б) усі весняні дати;
- в) саму пізню дату.

8. Дано символний файл f . Одержати файл g , утворений з файлу f заміною всіх його прописних (великих) літер однойменними рядковими (малими).

9. Дано файл f , що містить номери телефонів співробітників закладу. Указуються прізвище співробітника, його ініціали і номер телефону. Знайти телефон співробітника за його прізвищем та ініціалам.

10. Дано файл f , компоненти якого є цілими числами. Записати у файл g усі парні числа файлу f , а у файл h - усі непарні. Порядок проходження чисел зберігається.

11. Дано символний файл f . Записати у файл g всі слова файлу f в зворотному порядку.

12. Дано символні файли f і g . Записати у файл h спочатку компонента файлу f , потім компоненти файлу g зі збереженням порядку.

13. Дано файл f , компоненти якого є цілими числами. Одержати файл h , утворений з файлу f виключенням повторних входжень того самого числа.

14. Дано файл f , компоненти якого є цілими числами. Ніяка з компонентів файлу не дорівнює нулю. Файл f містить стільки ж від'ємних чисел, скільки і додатних. Використовуючи допоміжний файл h переписати компоненти файлу f у файл g так, щоб у файлі g :

а) не було двох сусідніх чисел з одним знаком;

б) спочатку йшли додатні, потім від'ємні числа;

в) числа йшли в наступному порядку: два додатних, два від'ємних, два додатних, два від'ємних і т.д. (передбачається, що число компонентів у файлі f ділиться на 4).

15. Дано файл f , компоненти якого є цілими числами. Ніяка з компонентів файлу f не дорівнює нулю. Числа у файлі йдуть у наступному порядку: десять додатних, десять від'ємних, десять додатних, десять від'ємних і т.д. Переписати компоненти файлу f у файл g так, щоб у файлі g числа йшли в наступному порядку:

а) п'ять додатних, п'ять від'ємних, п'ять додатних, п'ять від'ємних і т.д.;

б) двадцять додатних, двадцять від'ємних, двадцять додатних, двадцять від'ємних і т.д. (передбачається, що число компонентів файлу f ділиться на 40).

16. Дано файл f , компоненти якого є цілими числами. Число компонентів файлу ділиться на 10. Записати у файл g найбільше значення перших десяти компонентів файлу f , потім - наступних десяти компонентів, і т.д.

17. З умови попередньої задачі видаляється припущення про те, що число компонентів файлу f ділиться на 10. Якщо в останній групі виявиться менш десяти компонентів, то останній компонент файлу g повинен дорівнювати найбільшому з компонентів файлу f , що утворюють останню (неповну) групу.

18. Дано символічний файл f . Додати в його кінець символи E, N, D (якщо те необхідно, використовувати додатковий файл g).

19. Дано символічні файли f і g . Визначити, чи збігаються компоненти файлу f з компонентами файлу g . Якщо ні, то одержати номер першого компонента, у якому файли f і g відрізняються між собою. У випадку, коли один з файлів має n компонент ($n \geq 0$) і повторює початок іншого (більш довгого) файлу, відповіддю повинне бути число $n + 1$.

20. Дано символні файли f і g . Записати у файл h усі початкові співпадаючі компоненти файлів f і g .

21. Дано символний файл f . Записати у файл g зі збереженням порядку проходження ті символи файлу f :

- а) яким у цьому файлі передують букви "d"
- б) слідом за якими у цьому файлі йде буква "a".

22. Дано символний файл f . Групи символів, розділені пробілами (одним чи декількома), які не містять пробілів усередині себе, будемо надалі називати словами. Видалити з файлу всі однобуквені слова і зайві пробіли. Результат записати у файл g .

23. Дано символний файл f . Знайти саме довге слово (див. попередню задачу) серед слів, друга буква яких є "e"; якщо слів з найбільшою довжиною декілька, то знайти останнє. Якщо таких слів немає взагалі, то сповістити про це. Вирішити цю задачу:

- а) вважаючи, що слова складаються не більш ніж з 10-ти символів;
- б) без обмеження на число символів у слові.

24. Дано символний файл f . Вважаючи, що кількість символів у слові не перевершує двадцяти:

- а) визначити, скільки у файлі f міститься слів, що складаються з одного, двох, трьох і т.д. символів;
- б) одержати гістограму (стовпчасту діаграму) довжин усіх слів файлу f ;
- в) визначити кількість слів у файлі f .

25. Дано символний файл f . Передбачається, що довжина одного слова не перевершує десяти символів і що число слів ділиться на 100. Підготувати файл для друку слів у двох стовпчиках по п'ятдесят рядків на сторінці. Слова повинні бути розміщені у файлі в наступному порядку: 1-і слово, 51-і, 2-і, 52-і, ..., 50-і, 100-і, потім (наступна сторінка) 101-і, 151-і, ..., 150-і, 200-і і т.д.

26. Дано символний файл f , що містить відомості про співробітників установи, записані за наступним зразком: *прізвище ім'я по батькові, прізвище ім'я по батькові, ...* Записати ці відомості у файл g , використовуючи зразки:

- а) *ім'я по батькові прізвище, ім'я по батькові прізвище, ...* ;
- б) *прізвище і.пб., прізвище і.пб., ...*

27. Дано символний файл f , що містить довільний текст довжиною не більш 5000 слів. Слова в тексті розділені проміжками і розділовими знаками.

Одержати 10 слів, що найчастіше зустрічаються і число їхніх появ. Розв'язати задачу:

- а) без обмеження на довжини слів;
- б) припускаючи, що будь-яке слово тексту складається не більш ніж з 10 букв.

28. Дано два символних файли f і g . Файл f містить довільний текст. Слова в тексті розділені пробілами і розділовими знаками. Файл g містить не більш 40 слів, що розділені комами. Ці слова утворюють пари: кожне перше слово вважається замінім, кожне друге слово – таким, що заміняє. Знайти у файлі f всі замінні слова і замінити їх на відповідні, що замінюють. Результат помістити у файл h .

29. Пряма на площині задається рівнянням $ax + by + c = 0$, де a й b одночасно не дорівнюють нулю. Будемо розглядати тільки прямі, для яких коефіцієнти a , b і c - цілі числа. Нехай f - файл, що містить коефіцієнти декількох прямих (не менше трьох). Переписати з файлу f у файл g коефіцієнти тих прямих, що:

- а) рівнобіжні першій з прямих, заданих у файлі f ;
- б) зазначені в п. а), але додатково потрібно, щоб усі прямі були різні;
- в) перетинають першу з прямих, заданих у файлі f ;
- г) зазначені в п. в), але додатково потрібно, щоб серед прямих не було рівнобіжних.

30. Умова попередньої задачі обмежена. Потрібно одержати у файлі g коефіцієнти всіх різних прямих файлу f .

2.5 Задача 5. Табулювання функцій

1. Скласти схему алгоритму і програму обчислення значень функцій свого варіанта (див. табл.4) і вивести результати обчислень у вигляді таблиці.

2. Вказівки до виконання завдання.

– Дослідити методами математичного аналізу поведінку функції свого варіанта при заданих значеннях аргументу (табл.4).

– При наявності точок розриву передбачити в програмі відповідний захист.

– Результати обчислень вивести на екран у вигляді таблиці.

– Обчислення ведуться на інтервалі $x \in [0;3]$ з кроком 0,1. Додатково для всіх тригонометричних, гіперболічних та сполучених (стовпці 4, 6 та 8) функцій обчислити їх значення при $x = \frac{\pi}{3}; \frac{\pi}{4}; \frac{2\pi}{3}; \pi$.

Таблиця 4. Варіанти індивідуальних завдань.

N вар	Алгебраїчні раціональні	Алгебраїчні ірраціональні	Тригонометричні	Показові	Гіперболічні	Натуральні логарифми	Сполучення тригонометричних, гіперболічних і їх зворотних
-1-	--2--	--3--	--4--	--5--	--6--	--7--	--8--
1	$\frac{x^3}{1+x}$	$\frac{1}{x\sqrt{1+x}}$	$\frac{1-\sin x}{x}$	$e^{-(1-x)}$	$\frac{1}{sh^2 x}$	$\frac{1}{x \ln \frac{1+x}{x}}$	$sh(\sin x)$
2	$\frac{x}{(1-x)^3}$	$\frac{1+x^2}{\sqrt{1+x}}$	$\frac{1}{\sin^2 x(1+\sin x)}$	$\frac{1}{1+e^{-x}}$	$\frac{1}{x^2 shx}$	$\frac{1}{x^2 \ln(x+1)}$	$ch(\sin x)$
3	$x(x-1)^4$	$\frac{\sqrt{1+x}}{x^2}$	$\frac{(1-\sin x)^2}{\sin x}$	$\frac{1}{1-e^{-x}}$	$\frac{x^2}{chx}$	$\frac{1}{\ln(1+x^2)}$	$\varphi_1 = shx \sin x$
4	$\frac{x^3}{1-x}$	$\frac{\sqrt{(1+x)^3}}{x^2}$	$\frac{1}{\sin(1-x)}$	$\frac{x}{1+e^x}$	$\frac{x^2}{shx}$	$\frac{x-1}{\ln(x-1)}$	$\varphi_2 + \varphi_4$
5	$\frac{1}{(1+x^2)^2}$	$\frac{1}{x^3\sqrt{(1+x)^2}}$	$\frac{3}{\sin 3x}$	$\frac{e^x+1}{e^x-1}$	$\frac{1}{xsh^3 x}$	$\frac{1}{x(1-\ln x)}$	$shx + \sin x$
6	$\frac{x^2-1}{1+x^2}$	$\frac{\sqrt{1+x^2}}{x^2}$	$\frac{x}{\sin x - \sin^3 x}$	$\frac{x}{e^{-x}}$	$\frac{1}{(shx+1)^2}$	$\frac{1}{x^2 \ln(1+x^2)}$	$chx + \cos x$
7	$\frac{1}{x(1-x^2)}$	$\frac{1}{x\sqrt{x-1}}$	$\frac{1}{\sqrt{1+\sin^2 x}}$	$x^3 e^{-x}$	$\frac{shx-1}{x}$	$\frac{1}{x \ln(x^2-1)}$	$\frac{\sin x}{shx}$

-1-	--2--	--3--	--4--	--5--	--6--	--7--	--8--
8	$\frac{1+x^2}{x^2}$	$\frac{1}{x\sqrt{x^2-1}}$	$\frac{1}{x\sqrt{\sin x}}$	$\frac{x}{e^{4x}}$	$\frac{1}{sh\sqrt{x}}$	$\frac{1+x}{\ln(x^2-1)}$	$\frac{\sin x}{chx}$
9	$\frac{1-x^2}{x^2}$	$\frac{1}{\sqrt{x^4-1}}$	$\frac{1}{\sqrt{1-\sin^3 x}}$	$\frac{1}{xe^{\frac{x}{2}}}$	$\frac{1}{sh^3 x}$	$\frac{x^2}{\ln(x^2+1)}$	$\frac{1}{sh(\sin x)}$
10	$\frac{1}{x^2(1-x^4)}$	$\frac{x^2}{\sqrt{1+x^3}}$	$\frac{1}{\sqrt{1-\sin^3 x}}$	$\frac{1}{x^2 e^{x^2}}$	$\frac{1}{ch^{2x}}$	$\frac{x}{\ln^2(x-1)}$	$\frac{1}{ch(\sin x)}$
11	$\frac{1}{\frac{1}{2}-x^2}$	$\frac{1}{x\sqrt{1+x^2}}$	$\frac{1}{\sqrt[3]{\sin^2 x}}$	$\frac{1}{x^2 e^{x^3}}$	$\frac{1}{x^2 chx}$	$\frac{1}{\ln^2(1-x)}$	$\varphi_2 = shx \cos x$
12	$\frac{1}{(1-x^2)^3}$	$\frac{\sqrt{1+x^2}}{x^2 \sqrt{x}}$	$\frac{1}{\sqrt[4]{1-\sin^3 x}}$	$\frac{1}{e^{-x}(1-x^2)}$	$\frac{1}{(chx+1)^2}$	$\frac{x^2}{\ln(1-x)}$	$\frac{1}{\varphi_2 + \varphi_4}$
13	$\frac{1}{x^2(1+x^2)^2}$	$\frac{x^4}{\sqrt{1+x^2}}$	$\frac{1}{\sqrt[4]{1+\sin^2 x}}$	$\frac{1}{x^2 e^x}$	$\frac{chx-1}{x}$	$\frac{1-x}{\ln^2 x}$	$\frac{1}{shx + \sin x}$
14	$\frac{(1-x^2)^2}{x}$	$\frac{1}{\sqrt[3]{1-x}}$	$\frac{1}{\cos^2 x}$	$\frac{1}{x^2 e^{\frac{x^2}{2}}}$	$\frac{ch^2 x}{shx}$	$\frac{x}{\ln^2(1+x)}$	$\frac{1}{chx + \cos x}$
15	$\frac{1}{1-x^3}$	$\frac{1}{x^3 \sqrt[3]{1-x}}$	$\cos 3x$	$\frac{e^{x^3}}{e^{x^2}}$	$\frac{1}{ch2x}$	$\frac{1}{\ln \frac{x^2}{1+x}}$	$\frac{shx}{\sin x}$
16	$\frac{1+x^3}{x^2}$	$\frac{1}{x^2 \sqrt[3]{1-x}}$	$\frac{\cos x}{x}$	$\frac{1}{x^2 e^{\frac{x^2}{2}}}$	$\frac{1}{ch3x}$	$\frac{1}{\ln \frac{1+x}{x-1}}$	$\frac{chx}{\sin x}$
17	$\frac{1-x^3}{x^2}$	$\frac{x^2}{\sqrt[3]{1+x^2}}$	$\frac{1}{1+\cos^2 x}$	$\frac{1+e^{-x}}{e^{-x}}$	$\frac{x}{sh^3 x}$	$\frac{\sqrt{x+1}}{x \ln x}$	$sh(\cos x)$
18	$\frac{(1-x)^4}{x^2}$	$\frac{1}{\sqrt[3]{x^2-1}}$	$\frac{1}{1-\cos^2 x}$	$\frac{1}{e^{\sqrt{1+x}}}$	$\frac{1}{ch \frac{x}{2}}$	$\frac{\sqrt{x^2+1}}{x \ln x}$	$ch(\cos x)$
19	$\frac{1}{x^2(1-x^3)}$	$\sqrt{1-x^4}$	$\frac{x^2}{\cos^3 x}$	$\frac{1}{\sqrt{1+e^{-x}}}$	$\frac{1}{ch \frac{x}{4}}$	$\frac{\sqrt{x^2-1}}{x \ln x}$	$\varphi_4 = chx \sin x$
20	$\frac{1-x^3}{x(1+x^3)}$	$\frac{1}{x^3 \sqrt{(1-x)^2}}$	$\frac{1}{x^2 \cos^3 x}$	$\frac{\sqrt{x}}{e^{-x}}$	$\frac{1}{shx ch^2 x}$	$\frac{1}{x^2 \ln \frac{x}{\sqrt{x^2-1}}}$	$\varphi_4 - \varphi_2$
21	$\frac{1}{(1-x^3)^2}$	$\frac{1}{\sqrt[4]{1+x^2}}$	$\frac{1-\cos x}{x}$	$\frac{\sqrt{e^{-x}+1}}{e^{-x}}$	$\frac{1}{sh^3 x chx}$	$\frac{\sqrt{x}}{\ln \sqrt{x}}$	$shx - \sin x$
22	$\frac{1}{1-x^4}$	$\frac{1}{x^3 \sqrt[3]{(1-x)^2}}$	$\frac{(1-\cos x)^2}{\cos x}$	$\frac{1}{e^{-x} \sqrt{1+x^2}}$	$\frac{1}{shx(chx-1)}$	$\frac{1}{\ln \sqrt{1+x^2}}$	$chx - \cos x$
23	$\frac{1+x^4}{x^2}$	$\frac{\sqrt[3]{1-x}}{x^2}$	$\frac{1}{\cos x(1-\cos x)}$	$\frac{1}{\sqrt{2\pi} e^{-x}}$	$\frac{1+shx}{chx}$	$\frac{1}{x\sqrt{\ln x}}$	$\frac{\cos x}{shx}$

-1-	--2--	--3--	--4--	--5--	--6--	--7--	--8--
24	$\frac{1-x^4}{x^2}$	$\frac{x^2}{\sqrt[3]{1+x^4}}$	$\frac{1}{\sqrt{\cos x}}$	$\frac{1}{e^x \sqrt[3]{x^2}}$	$\frac{1}{th^2 x}$	$x^2 \ln \frac{1+x}{x}$	$\frac{1}{sh(\cos x)}$
25	$\frac{1+x}{x^3}$	$\frac{x^2}{\sqrt[4]{1+x^2}}$	$\frac{\sin x}{1+\sin x}$	$\frac{1}{1+x^4 e^{-\frac{x^2}{2}}}$	$\frac{x}{th^2 x}$	$x^3 \ln(x+1)$	$\frac{1}{ch(\cos x)}$
26	$\frac{x}{(1+x)^4}$	$\sqrt{\sin x}$	$\sqrt{\sin x}$	$\frac{1}{e^{\cos x}}$	$sh^3 x$	$\ln(1+x^3)$	$\varphi_3 = chx \cos x$
27	$x(x-1)^5$	$(1+x)^2 \sqrt{1+x}$	$1+\sin(x+1)$	$\frac{1}{e^{-x} \sin x}$	$x^3 shx$	$\frac{\ln(x^2+1)}{x^2+1}$	$\frac{1}{\varphi_4 - \varphi_2}$
28	$\frac{x^3}{(1+x)^2}$	$\frac{x^3}{\sqrt{1+x}}$	$\sin(\sin x)$	$\frac{1}{e^{\cos^2 x}}$	$\frac{shx}{x^3}$	$\frac{1+\ln x}{1-\ln x}$	$\frac{1}{shx - \sin x}$
29	$1-x^2$	$\frac{x^3}{\sqrt{(1+x)^3}}$	$\frac{\sin 4x}{4}$	e^{1+x^2}	$xch^2 x$	$x^3 \ln(1+x^2)$	$\frac{1}{chx - \cos x}$
30	$\frac{1-x^2}{x-1}$	$\frac{x^3}{\sqrt{1-x}}$	$\frac{1+\sin x}{x^2}$	$\frac{1-e^{-x}}{x(1+e^{-x})}$	$shx - 1$	$x^2 \ln(x^2 - 1)$	$\frac{chx}{\cos x}$

2.6 Задача 6. Класи в Python

1. Розробити клас та продемонструвати результати його тестування.
2. Вказівки до виконання завдання.
 - Розробити базовий клас та клас-спадкоємець відповідно варіанта (табл.4)
 - Створіть методи класу, які зазначені в умові завдання.
 - Зробіть необхідні тести до модуля.

Таблиця 5. Таблиця розподілу варіантів

Завдання 1	Завдання 2	Завдання 3	Завдання 4
Варіант 1, 5, 9, 13, 17, 21, 25	Варіант 2, 6, 10, 14, 18, 22, 26	Варіант 3, 7, 11, 15, 19, 23, 27	Варіант 4, 8, 12, 16, 20, 24, 28

Завдання 1.

Створіть клас Alphabet. Його метод `__init__()`, буде мати визначені два параметри: `lang` - мова і `letters` - список букв. Значення змінних `lang` і `letters` будуть визначені за замовчуванням і міститимуться у вигляді статичних атрибутів для української мови. Клас матиме метод `print_alphabet()`, який виведе в консоль літери українського алфавіту. Метод `letters_num()`, повертатиме кількість букв в алфавіті. Метод `is_ua_lang()` прийматиме довільний текст і визначатиме чи відноситься він до української мови (незалежно від регістру).

Створіть клас EngAlphabet шляхом успадкування від класу Alphabet. Для його методу `__init__()`, всередині якого буде викликатися батьківський метод `__init__()`, в якості параметрів будуть передаватися позначення мови

(наприклад, 'En') і рядок, що складається з усіх букв алфавіту. Додайте приватний статичний атрибут `__en_letters_num`, який буде зберігати кількість букв в алфавіті. Створіть метод `is_en_letter()`, який буде приймати строку в якості параметра і визначати, чи відноситься цей рядок до англійського алфавіту. Перевизначити метод `letters_num()` - нехай в поточному класі він буде повертати значення властивості `__en_letters_num`. Створіть статичний метод `example()`, який буде повертати приклад тексту англійською мовою.

Тести до завдання 1

- Створіть об'єкт класу `EngAlphabet`
- Надрукуйте літери алфавіту для цього об'єкту
- Виведіть кількість букв в алфавіті
- Перевірте, чи відноситься буква J до англійського алфавіту.
- Перевірте, чи відноситься буква Щ до українського алфавіту
- Виведіть приклад тексту англійською мовою

Завдання 2.

Створіть клас `Human`. Визначте для нього два статичних атрибути: `default_name` і `default_age`. Його метод `init()`, який крім `self` приймає ще два публічних параметри (`name` і `age`)

і два приватних (`money` і `house`). Параметр `money` визначатиме кількість грошей, а `house` - посилання на об'єкт класу `House`. Метод `info()`, має виводити поля `name`, `age`, `house` і `money`. Реалізуйте довідковий статичний метод `default_info()`, який буде виводити статичні поля `default_name` і `default_age`. Реалізуйте приватний метод `make_deal()`, який буде відповідати за технічну реалізацію покупки будинку: зменшувати кількість грошей на рахунок і привласнювати посилання на тільки що куплений будинок. В якості аргументів даний метод приймає об'єкт будинку та його ціну. Реалізуйте метод `earn_money()`, що збільшує значення поля `money`. Реалізуйте метод `buy_house()`, який буде перевіряти, що у людини достатньо грошей для покупки, і здійснювати операцію. Якщо грошей занадто мало - потрібно вивести попередження в консоль. Параметри методу: посилання на будинок і розмір знижки (за замовчуванням 10%).

Створіть клас `House`. Його метод `__init__()` містить два динамічних параметри: `_area` і `_price`, що мають значення за замовчуваннями. Створіть метод `final_price()`, який приймає як параметр розмір знижки і повертає ціну з урахуванням даної знижки.

Створіть клас `SmallHouse`, успадкувавши його функціонал від класу `House`. Всередині класу `SmallHouse` перевизначите метод `__init__()` так, щоб він створював об'єкт з площею 40 м².

Тести до завдання 2

- Викличте довідковий метод `default_info()` для класу `Human`
- Створіть об'єкт класу `Human`
- Виведіть довідкову інформацію про створений об'єкт (викличте метод

info ()).

- Створіть об'єкт класу SmallHouse
- Спробуйте купити створений будинок, переконайтеся в отриманні попередження.
- виправте фінансове становище об'єкта - викличте метод `earn_money()`
- Знов спробуйте купити будинок
- Подивіться, як змінився стан об'єкта класу Human.

Завдання 3.

Створіть клас Apple. Його статичний атрибут `states`, яке буде містити всі стадії дозрівання яблука («Відсутнє», «Цвітіння», «Зелене», «Червоне»). Метод `__init__()`, всередині якого будуть визначені два динамічних `protected` атрибути: `_index` (номер яблука) і `_state` (приймає перше значення зі словника `states`). Створіть метод `grow()`, який буде переводити яблуко на наступну стадію дозрівання. Створіть метод `is_ripe()`, який буде перевіряти, що яблуко дозріло (досягло останньої стадії дозрівання).

Створіть клас `AppleTree`. Визначте метод `__init__()`, який буде приймати як параметр кількість яблук і на його основі буде створювати список об'єктів класу `Apple`. Даний список буде зберігатися всередині динамічного атрибуту `apples`. Створіть метод `grow_all()`, який буде переводити всі об'єкти зі списку яблук на наступний етап дозрівання.

Створіть метод `all_are_ripe()`, який буде повертати `True`, якщо всі яблука зі списку стали стиглими. Створіть метод `give_away_all()`, який буде чистити список яблук після збору врожаю.

Створіть клас `Gardener`. Його метод `__init__()`, міститиме два динамічних атрибути: `name` (ім'я садівника, публічний атрибут) і `_tree` (приймає об'єкт класу `AppleTree`). Створіть метод `work()`, який змушує садівника працювати, що дозволяє яблукам ставати більш стиглими. Створіть метод `harvest()`, який перевіряє, чи всі плоди дозріли. Якщо всі - садівник збирає урожай. Якщо і - метод друкує попередження. Створіть статичний метод `apple_base()`, який виведе в консоль довідку з кількості яблук і ступені їх стиглості.

Тести до завдання 3

- Створіть декілька об'єктів класу `Apple`.
- Викличте довідку по всім наявним яблукам
- Створіть об'єкти класів `AppleTree` і `Gardener`
- Використовуючи об'єкт класу `Gardener`, попрацювати над яблучним деревом.
- Спробуйте зібрати урожай
- Якщо яблука ще не дозріли, продовжуйте доглядати за деревом
- Зберіть урожай.

Завдання 4.

Створіть клас `KmrCsv`, який має два атрибути класу за замовчуванням: `ref` (посилання на CSV файл з оцінками) і `num` (номер КМР), та методи для

встановлення і, відповідно, визначення посилання на файл з оцінками, встановлення номеру КМР, читання файлу з оцінками та виведення інформації про файл (номер КМР і кількість студентів, що її виконали).

Створіть клас `Statistic`, що містить наступні методи:

- `avg_stat()` визначає відсотки правильних відповідей на кожне питання серед усіх студентів і повертає результат у вигляді кортежу чисел;
- метод `marks_stat()` визначає яку оцінку набрала відповідна кількість студентів і повертає результат у вигляді словника формату {оцінка: кількість студентів};
- метод `marks_per_time()` визначає, який середній бал за хвилину набрав студент під час виконання КМР і повертає результат у вигляді словника формату {id студента (це перша колонка csv файлу): середній бал за хвилину} ;
- метод `best_marks_per_time()`, який приймає два аргументи `bottom_margin` і `top_margin` (нижня і верхня межа вибірки підсумкових балів за КМР), та формує для цієї вибірки п'ять найкращих результатів середніх балів за хвилину у вигляді кортежу формату (id студента, підсумкова оцінка, середній бал за хвилину).

Створіть клас `Plots`, що містить наступні методи:

- `set_cat()` - встановлює каталог, в який зберігатимуться отримані графіки;
- `avg_plot()` - приймає кортеж з відсотками правильних відповідей на кожне окреме питання, формує гістограму на його основі і зберігає отриманий графік;
- `marks_plot()` - приймає словник з оцінками і кількістю студентів, що їх набрали, формує на його основі гістограму і зберігає її;
- `best_marks_plot()` - формує для п'яти найкращих результатів середніх балів за хвилину гістограму і зберігає її.

Створіть клас `KmrWork`, що успадковує класи `CsvKmr`, `Statistic` і `Plots`. В якості аргументів екземпляр класу приймає посилання на csv файл та номер КМР.

Клас `KmrWork` містить наступні статичні атрибути

- `kmrs` - в ньому зберігається словник формату {номер КМР: адреса відповідного csv файлу}
- `cat` - каталог для збереження результатів роботи

Крім успадкованих, клас `KmrWork` містить наступні методи:

- `compare_csv()` - виводить на екран і зберігає в txt файл результат порівняння статистики двох КМР (кількість виконаних КМР, середній бал за КМР, середній час виконання КМР);
- `compare_avg_plots()` - виводить на екран і зберігає дві гістограми з відсотками правильних відповідей на кожні окремі питання.

Тести до завдання 4

- Створіть об'єкти `kmr1` і `kmr2` класу `KmrWork`.
- Використайте для об'єкту `kmr2` методи `avg_plot()` і `marks_plot()`
- Для класу `KmrWork` використайте методи `compare_csv()` і

compare_avg_plots().

3. ВИМОГИ ДО СТРУКТУРИ, ЗМІСТУ ТА ОФОРМЛЕННЯ ЗВІТУ

Рекомендується, щоб звіт з практики навчальної комп'ютерної включав:

- титульну сторінку (форма наведена в додатку 1);
- зміст;
- завдання, які були виконані під час проходження практики;
- кожне завдання має обов'язково містити наступну інформацію: умова задачі відповідно до номеру варіанту, програмний код, який виконує поставлену задачу, результати тестування завдань;
- висновки, в яких характеризуються навички та знання, що були отримані під час виконання практики;
- перелік джерел інформації, що були використані під час виконання завдань;

Звіт з практики має наступну структуру (табл. 6).

Таблиця 6

Загальна структура звіту з передатестаційної практики

Назва структурної частини	Рекомендований обсяг, сторінок
Титульна сторінка	1
Зміст	1
Задачі, що було виконано на практиці	10 - 20
Висновки	1 - 2
Список використаних джерел	не менше 5 найменувань

Звіт оформлюється за допомогою комп'ютера відповідно до загальних правил.

Мова і стиль звіту. Звіт з практики виконується державною мовою. Мовностилістичне оформлення обумовлюється особливостями наукового стилю, враховуючи:

- формально-логічний спосіб викладення, смислової завершеності, цілісності та зв'язності думок;
- відсутність емоційного забарвлення наукового тексту;
- об'єктивність викладу, смислової точності, стислості, ясності;
- використання спеціальної термінології з посиланням на авторитетні джерела (підручники, довідкові та нормативні видання тощо).

Основні технічні вимоги до оформлення звіту:

- текстовий редактор – MS Word 2003 і вище;
- міжрядковий інтервал – 1,5,
- кегель Times New Roman – 14,
- абзацний виступ – 1,25 см,
- поля зверху та знизу – 2 см, ліворуч – 3 см, праворуч – 1,5 см.
- при використанні в тексті рисунків або скріншотів всі вони мають супроводжуватись підписом, який розкриває його сенс.

Звіт з навчальної комп'ютерної практики оцінюється за 100-бальною шкалою. Якщо звіт, представлений на кафедру для перевірки, не відповідає вище зазначеним вимогам, він може бути повернений на доопрацювання без коригування термінів здачі та захисту.

На титульній сторінці звіту зазначається дата його подачі на перевірку. Термін перевірки залежить від сукупного обсягу наданих на перевірку робіт. Мінімальний термін перевірки – тиждень з моменту подачі на кафедру.

Захист звіту – це усна бесіда керівника зі здобувачем з приводу недоліків роботи та питань, що виникли в процесі перевірки, в ході якої здобувач має змогу дати відповіді на питання, аргументовано відстояти свою реалізацію задач і отримати бали (структура оцінки наведена в табл. 7.).

Таблиця 7

Система критеріїв оцінювання звіту та розподіл балів

Критерії оцінки	Допустима кількість балів (max)
Завдання 1	12
Супровід послідовності дій, використаних команд та змінних під час виконання завдання коментарями. Програма повністю виконує поставлене завдання	8
Супровід функції розрахунку функціонального ряду блок-схемою	4
Завдання 2	12
Супровід послідовності дій, використаних команд та змінних під час виконання завдання коментарями. Програма повністю виконує поставлене завдання	8
Супровід програми блок-схемою, що розкриває суть коду	4
Завдання 3	12
Супровід послідовності дій, використаних команд та змінних під час виконання завдання коментарями. Програма повністю виконує поставлене завдання	8
Супровід програми блок-схемою, що розкриває суть коду	4
Завдання 4	12
Супровід послідовності дій, використаних команд та змінних під час виконання завдання коментарями. Програма повністю виконує поставлене завдання	8
Супровід програми блок-схемою, що розкриває суть коду	4
Завдання 5	12
Супровід послідовності дій, використаних команд та змінних	8

під час виконання завдання коментарями. Табуляція функцій виконується з урахуванням особливостей ОДЗ функцій	
Реалізовано побудову графіків з можливістю вибору номера функції	4
Завдання 6	12
Супровід класів, методів та їх параметрів коментарями. Клас повністю відповідає вимогам завдання	8
Супровід класу необхідними тестами, що розкривають функціональність	4
Оформлення звіту	20
відповідає вимогам	20
частково відповідає вимогам	15
не відповідає вимогам	5
Терміни здачі звіту	8
своєчасно (протягом двох тижнів після закінчення)	8
із затримкою	2
Загальна кількість балів	100

4. КРИТЕРІЇ ОЦІНЮВАННЯ

Проходження практики оцінюється керівником за допомогою критеріїв, що характеризують співвідношення вимог до рівня компетентностей і показників оцінки за рейтинговою шкалою. Зміст критеріїв спирається на компетентнісні характеристики, визначені НРК для бакалаврського рівня вищої освіти (подано нижче).

Опис кваліфікаційного рівня	Вимоги до знань, умінь/навичок, комунікації, відповідальності і автономії	Показник оцінки
<i>Знання</i>		
♦ концептуальні наукові та практичні знання, критичне осмислення теорій, принципів, методів і понять у сфері професійної діяльності та/або навчання	Відповідь відмінна – правильна, обґрунтована, осмислена. Характеризує наявність: - концептуальних знань; - високого ступеню володіння станом питання; - критичного осмислення основних теорій, принципів, методів і понять у навчанні та професійній діяльності	95-100
	Відповідь містить негрубі помилки або описки	90-94
	Відповідь правильна, але має певні неточності	85-89
	Відповідь правильна, але має певні неточності й недостатньо обґрунтована	80-84
	Відповідь правильна, але має певні неточності, недостатньо обґрунтована та осмислена	74-79
	Відповідь фрагментарна	70-73
	Відповідь демонструє нечіткі уявлення студента про об'єкт вивчення	65-69
	Рівень знань мінімально задовільний	60-64
	Рівень знань незадовільний	<60

Опис кваліфікаційного рівня	Вимоги до знань, умінь/навичок, комунікації, відповідальності і автономії	Показник оцінки
Уміння/навички		
<p>♦ поглиблені когнітивні та практичні уміння/навички, майстерність та інноваційність на рівні, необхідному для розв'язання складних спеціалізованих задач і практичних проблем у сфері професійної діяльності або навчання</p>	<p>Відповідь характеризує уміння:</p> <ul style="list-style-type: none"> - виявляти проблеми; - формулювати гіпотези; - розв'язувати проблеми; - обирати адекватні методи та інструментальні засоби; - збирати та логічно й зрозуміло інтерпретувати інформацію; - використовувати інноваційні підходи до розв'язання завдання 	95-100
	<p>Відповідь характеризує уміння/навички застосовувати знання в практичній діяльності з негрубими помилками</p>	90-94
	<p>Відповідь характеризує уміння/навички застосовувати знання в практичній діяльності, але має певні неточності при реалізації однієї вимоги</p>	85-89
	<p>Відповідь характеризує уміння/навички застосовувати знання в практичній діяльності, але має певні неточності при реалізації двох вимог</p>	80-84
	<p>Відповідь характеризує уміння/навички застосовувати знання в практичній діяльності, але має певні неточності при реалізації трьох вимог</p>	74-79
	<p>Відповідь характеризує уміння/навички застосовувати знання в практичній діяльності, але має певні неточності при реалізації чотирьох вимог</p>	70-73
	<p>Відповідь характеризує уміння/навички застосовувати знання в практичній діяльності при виконанні завдань за зразком</p>	65-69
	<p>Відповідь характеризує уміння/навички застосовувати знання при виконанні завдань за зразком, але з неточностями</p>	60-64
<p>рівень умінь/навичок незадовільний</p>	<60	
Комунікація		
<p>♦ донесення до фахівців і нефахівців інформації, ідей, проблем, рішень, власного досвіду та аргументації;</p> <p>♦ збір, інтерпретація та застосування даних;</p> <p>♦ спілкування з професійних питань, у тому числі іноземною мовою, усно та письмово</p>	<p>Вільне володіння проблематикою галузі. Зрозумілість відповіді (доповіді). Мова:</p> <ul style="list-style-type: none"> - правильна; - чиста; - ясна; - точна; - логічна; - виразна; - лаконічна. <p>Комунікаційна стратегія:</p> <ul style="list-style-type: none"> - послідовний і несуперечливий розвиток думки; - наявність логічних власних суджень; - доречна аргументації та її відповідність відстоюваним положенням; - правильна структура відповіді (доповіді); - правильність відповідей на запитання; - доречна техніка відповідей на запитання; - здатність робити висновки та формулювати пропозиції 	95-100
	<p>Достатнє володіння проблематикою галузі з незначними хибами.</p>	90-94

Опис кваліфікаційного рівня	Вимоги до знань, умінь/навичок, комунікації, відповідальності і автономії	Показник оцінки
	Достатня зрозумілість відповіді (доповіді) з незначними хибами. Доречна комунікаційна стратегія з незначними хибами	
	Добре володіння проблематикою галузі. Добра зрозумілість відповіді (доповіді) та доречна комунікаційна стратегія (сумарно не реалізовано три вимоги)	85-89
	Добре володіння проблематикою галузі. Добра зрозумілість відповіді (доповіді) та доречна комунікаційна стратегія (сумарно не реалізовано чотири вимоги)	80-84
	Добре володіння проблематикою галузі. Добра зрозумілість відповіді (доповіді) та доречна комунікаційна стратегія (сумарно не реалізовано п'ять вимог)	74-79
	Задовільне володіння проблематикою галузі. Задовільна зрозумілість відповіді (доповіді) та доречна комунікаційна стратегія (сумарно не реалізовано сім вимог)	70-73
	Часткове володіння проблематикою галузі. Задовільна зрозумілість відповіді (доповіді) та комунікаційна стратегія з хибами (сумарно не реалізовано дев'ять вимог)	65-69
	Фрагментарне володіння проблематикою галузі. Задовільна зрозумілість відповіді (доповіді) та комунікаційна стратегія з хибами (сумарно не реалізовано 10 вимог)	60-64
	Рівень комунікації незадовільний	<60
Відповідальність і автономія		
<ul style="list-style-type: none"> ◆ управління складною технічною або професійною діяльністю чи проектами; ◆ спроможність нести відповідальність за вироблення та ухвалення рішень у непередбачуваних робочих та/або навчальних контекстах; ◆ формування суджень, що враховують соціальні, наукові та етичні аспекти; ◆ організація та керівництво професійним розвитком осіб та груп; ◆ здатність продовжувати навчання із значним ступенем автономії 	<p>Відмінне володіння компетенціями менеджменту особистості, орієнтованих на:</p> <ol style="list-style-type: none"> 1) управління комплексними проектами, що передбачає: <ul style="list-style-type: none"> - дослідницький характер навчальної діяльності, позначена вмінням самостійно оцінювати різноманітні життєві ситуації, явища, факти, виявляти і відстоювати особисту позицію; - здатність до роботи в команді; - контроль власних дій; 2) відповідальність за прийняття рішень в непередбачуваних умовах, що включає: <ul style="list-style-type: none"> - обґрунтування власних рішень положеннями нормативної бази галузевого та державного рівнів; - самостійність під час виконання поставлених завдань; - ініціативу в обговоренні проблем; - відповідальність за взаємовідносини; 3) відповідальність за професійний розвиток окремих осіб та/або груп осіб, що передбачає: <ul style="list-style-type: none"> - використання професійно-орієнтованих навичок; - використання доказів із самостійною і правильною аргументацією; - володіння всіма видами навчальної діяльності; 4) здатність до подальшого навчання з високим рівнем автономності, що передбачає: <ul style="list-style-type: none"> - ступінь володіння фундаментальними знаннями; 	95-100

Опис кваліфікаційного рівня	Вимоги до знань, умінь/навичок, комунікації, відповідальності і автономії	Показник оцінки
	<ul style="list-style-type: none"> - самостійність оцінних суджень; - високий рівень сформованості загальнонавчальних умінь і навичок; - самостійний пошук та аналіз джерел інформації 	
	Упевнене володіння компетенціями менеджменту особистості (не реалізовано дві вимоги)	90-94
	Добре володіння компетенціями менеджменту особистості (не реалізовано три вимоги)	85-89
	Добре володіння компетенціями менеджменту особистості (не реалізовано чотири вимоги)	80-84
	Добре володіння компетенціями менеджменту особистості (не реалізовано шість вимог)	74-79
	Задовільне володіння компетенціями менеджменту особистості (не реалізовано сім вимог)	70-73
	Задовільне володіння компетенціями менеджменту особистості (не реалізовано вісім вимог)	65-69
	Рівень відповідальності і автономії фрагментарний	60-64
	Рівень відповідальності і автономії незадовільний	<60

РЕКОМЕНДОВАНА ЛІТЕРАТУРА

1. Кормен, Томас Г. Вступ до алгоритмів : Переклад з англійської третього видання : [укр.] Introduction to Algorithms: Third Edition : [пер. з англ.] / Томас Г. Кормен, Чарлз Е. Лейзерсон, Роналд Л. Рівест, Кліфорд Стайн, – Київ : К. І. С., 2019. – 1288 с.

2. Трінтіна Н.А., Негоденко О.В., Гаманюк І.М., Шевченко С.М. Програмування мовою Python. Навчальний посібник підготовлено до друку для самостійної роботи студентів вищих навчальних закладів. – Київ : ННІТ ДУТ, 2022. – 113 с.

3. Копей В. Б. Мова програмування Python для інженерів і науковців : навч. посіб. / В. Б. Копей. – Івано-Франківськ : ІФНТУНГ, 2019. – 272 с.

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

5. Крєневич А.П. Python у прикладах і задачах. Частина 2. Об'єктно-орієнтоване програмування. Навчальний посібник – Київ : ВПЦ "Київський Університет", 2020. – 152 с.

6. Основи програмування. Python. Частина 1 [Електронний ресурс]: підручник для студ. спеціальності 122 "Комп'ютерні науки" /А.В. Яковенко; КПІ ім. Ігоря Сікорського. – Київ: КПІ ім. Ігоря Сікорського, 2018. – 195 с.

7. Руденко В., Жугастров О. Інформатика. Основи алгоритмізації та програмування мовою Python. Харків: Ранок, 2019. – 192 с.

Інформаційні ресурси в Інтернеті

1. <https://www.python.org/>
2. <https://docs.python.org/3/index.html>
3. <https://docs.python.org/3/library/index.html>
4. [https:// www.w3schools.com/python/default.asp](https://www.w3schools.com/python/default.asp)

ДОДАТОК 1 Зразок оформлення титульної сторінки

Міністерство освіти і науки України
Національний технічний університет
«Дніпровська політехніка»



Кафедра системного аналізу та управління

ЗВІТ З ПРАКТИКИ НАВЧАЛЬНОЇ КОМП'ЮТЕРНОЇ

Варіант: _____

Студента (ки) групи _____

(шифр групи, прізвище та ініціали)

(підпис)

Керівник практики: _____

(Прізвище та ініціали, шифр групи)

(підпис)

Дніпро – 2025

Навчальне видання

Хом'як Тетяна Валеріївна
Желдак Тимур Анатолійович
Гаранжа Дмитро Миколайович

ПРАКТИКА НАВЧАЛЬНА КОМП'ЮТЕРНА

Методичні рекомендації
для здобувачів ступеня бакалавра
освітньо-професійної програми «Системний аналіз»
зі спеціальності 124 Системний аналіз

Видано в авторській редакції.
Підписано до видання 05.09.2024. Авт. арк. 1,56.

Національний технічний університет «Дніпровська
політехніка».
49005, м. Дніпро, просп. Дмитра Яворницького, 19.