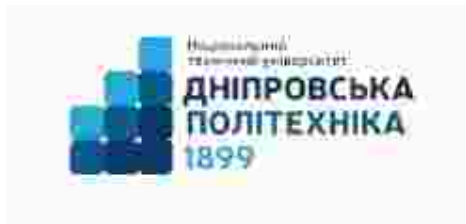


Міністерство освіти і науки України
НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ
«ДНІПРОВСЬКА ПОЛІТЕХНІКА»



Кафедра економіки та економічної кібернетики

І.М. Пістунів.

МОВА ПРОГРАМУВАННЯ PУTОН
МЕТОДИЧНІ РЕКОМЕНДАЦІЇ ДО ВИКОНАННЯ ПРАКТИЧНИХ РОБІТ
для студентів загальноекономічної підготовки

Дніпро
2024

Мова програмування Python: метод. рекомендації до виконання практичних робіт для студентів спеціальності 051 «Економіка» / уклад. І.М. Пістунов. – Дніпро: НТУ «ДП», 2024. – 26 с.

Укладач **І.М. Пістунов**, докт. техн. наук, проф.

ЗМІСТ

Методичні вказівки до виконання практичних завдань	4
Завдання №1. . Банківська позика.....	5
Завдання №2. Логічні оператори, булевий тип даних	6
Завдання №3. Конкатенація і повторення ряд.....	7
Завдання №4. Обмін значень змінних	8
Завдання № 5. Програма "Вгадай число від 1 до N"	9
Завдання №6. Робота з файлами	9
Завдання №7. Визначити індекси елементів масиву (списку), значення яких належать заданому діапазону	10
Завдання №8. Малювання з використанням черепашої графіки.....	10
Завдання №9. Робота з списками в Python	12
Завдання №10. Створення GUI додатку.....	12
Завдання № 11. Знайдення оптимального рішення.....	16
Екзаменаційне завдання. Робота з комплексом Python в Інтернеті	24

Методичні вказівки до виконання практичних завдань

Навчальним планом з дисципліни "Мова програмування Python" передбачено виконання практичних робіт.

Перед початком програмування необхідно вивчити відповідний розділ теоретичного матеріалу. При виконанні практичної роботи студент повинний дотримувати таких правил:

1. Готова програма розміщується у папці з вашим прізвищем на «хмарі».
2. Програми мають містити інформацію про автора, номер групи та дату виконання.
3. Кожен скрипт (оператор, команда) необхідно супроводжувати поясненнями у вигляді тексту чи формул.
4. Якщо практична робота після перевірки не зарахована, треба виправити помилки згідно з зауваженнями викладача. Доопрацьована практична робота надається для повторної перевірки разом з першим варіантом.
5. Студент, що не виконав практичні роботи, до іспиту не допускається.

Індивідуальне завдання №1. Банківська позика

Потрібно розрахувати вартість кредиту, тобто треба обчислити, скільки доведеться платити в місяць по позиці та скільки всього віддати грошей банку за весь період.

Місячна виплата по позиці обчислюється за такою формулою:

$$m = (s * p * (1 + p)^n) / (12 * ((1 + p)^n - 1)).$$

де:

m – розмір місячної виплати;

s – сума позики (кредиту) (№ залікової книжки);

p – відсоток банку, виражений в частках одиниці (тобто якщо 20%, то буде 0.2)
(поточний рік-2000).

n – кількість років, на які береться позика (№ по списку у журналі).

Параметри s , p та n – вводяться користувачем.

Результат: m – розмір місячної виплати; sum – загальна сума за весь період.

Округлити неціле число до двох знаків після коми.

Індивідуальне завдання №2. Логічні оператори, булевий тип даних

Напишіть програму, що визначає:

- 1) який з двох введених рядків довший,
- 2) чи введено порожній рядок,
- 3) чи рядки однакові,
- 4) яке з двох введених чисел більше,
- 5) чи буде від'ємною сума введених чисел,

за допомогою логічних функцій.

Користувачу подається запит, наприклад:

```
Введіть перший рядок: tu
Введіть другий рядок: kl;
Введіть перше число: 6.8
Введіть друге число: 8
-----
Перший рядок більший за другий: True
Числа не рівні: True
Перший рядок більший другого та числа не рівні: True
Сума чисел більше нуля АБО ні один з рядків не пустий: True
```

В Python є прості логічні оператори (`=`, `!=`, `<`, `>`, `<=`, `>=`) і складні (*and*, *or*, *not*). Всі логічні оператори, за винятком *not*, є бінарними. Це означає, що зліва і праворуч від них повинні стояти вирази. За допомогою логічних операторів ці вирази так чи інакше порівнюються між собою.

Результат логічних операцій має булевий тип даних (вбудований *class 'bool'* в Python), тобто може приймати лише два значення – “істина” та “неправда”. Потрібно бути обережним, порівнюючи між собою різні типи даних, так як це не завжди можливо. Наприклад, не можна порівнювати числа і рядки, але дробові та цілі числа - можна.

У складних логічних виразах потрібно враховувати послідовність операцій. Якщо немає впевненості, яка операція має пріоритет, то краще використовувати дужки.

Індивідуальне завдання №3. Конкатенація і повторення рядків

Обов'язкова наявність коментарів з умовами завдання перед написанням коду.

В Python над двома рядками можна виконати операцію, що позначається знаком `+`. Однак, на відміну від чисел, виконується не складанням (що для рядків в принципі неможливо), а з'єднання, тобто до кінця першого рядка додається другий. По-іншому операція з'єднання рядків називається *конкатенацією*.

Крім того, в Python є операція повторення (мультиплікації) рядків. Вона позначається знаком `*` (також як операція множення для чисел). При повторенні рядки з одного боку від знаку `*` ставиться рядок, а з іншого число, що позначає кількість повторів. При цьому не важливо, який об'єкт з якого боку знаходиться (зліва від знаку можна писати число, а праворуч – рядок).

В одному вираженні можна поєднувати операції конкатенації і мультиплікація. При цьому більш високий пріоритет у операції повторення рядка.

Напишіть програму, в якій:

- 1) Користувач вводить два рядка та кількість повторів (Прізвище, Ім'я, По-батькові) та шифр групи,
- 2) Виконується конкатенація рядків (Прізвище, Ім'я, По-батькові+ “студент/ка групи”+шифр групи),
- 3) Пошук підрядка в рядку,
- 4) Повтор об'єднаних рядків задану кількість раз, з нового рядка;
- 5) Надрукувати символ “*” задану кількість раз у вигляді:

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

- 6) Надрукувати ініціали ПІБ, наприклад: Михайло Сергійович Нестеренко

М		М	С	С	С	С	С	Н		Н
М	М		М	М	С			Н		Н
М		М		М	С			Н	Н	Н
М			М		С			Н		Н
М			М		С	С	С	С	С	С

Індивідуальне завдання №4. Обмін значень змінних

Обмін значень двох змінних – це "дія", в результаті якого одна змінна приймає значення, рівне другою змінною величиною, а друга – першої. Припустимо, є дві змінні a і b . При цьому $a = 5$ і $b = 6$. В результаті обміну має стати $a = 6$ і $b = 5$.

Зрозуміло, що якщо спробувати зробити такий обмін "по простому", тобто спочатку першій змінній привласнити значення другої, а другій – значення першої, то нічого не вийде. Якщо виконати вираз $a = b$, то змінна a буде посилатися на число 6, також як і b . Число 5 буде втрачено, так як на нього вже не буде посилатися жодна змінна, і вираз $b = a$ безглуздо, так як b буде присвоєно його ж поточне значення (6 в даному випадку).

Тому в багатьох мовах програмування (наприклад, Pascal) доводиться вводити третю змінну, що грає роль буфера (її іноді називають буферною змінною). У цій змінній зберігають значення першої змінної, потім першій змінній привласнюють значення другої, в нове значення для другої змінної беруть з буфера. Існує і інший варіант обміну без додаткової змінної, реалізуйте і його.

Також необхідно реалізувати обмін значень через кортежі.

Індивідуальне завдання № 5. Програма "Вгадай число від 1 до N"

Користувач спочатку вводить число загадує число від 1 до N.

Потрібно його відгадати, задавши якомога менше питань, і обмежитися використання оператора розгалуження та операторів циклу.

Щоб користувачеві задати менше питань, треба "розділити" діапазон чисел на дві по можливості рівні частини і визначити, в якій із них знаходиться шукане число.

Індивідуальне завдання № 6. Робота з файлами

Тема: запис/зчитування даних в/з текстовий файл, *.csv файл.

- Робота з файлами:
 - 1) Створити файл *Resume.txt*
 - 2) Ввести у файл інформацію про претендента: ПІБ, кваліфікацію, телефон.
 - 3) Закрити файл *Resume.txt*
 - 4) Вивести на екран інформацію з файлу *Resume.txt*

- Скласти картотеку з файлів зі структурою:
 - 5) Запросити у користувача ввести шлях до файлу (*user_url*).
 - 6) Запросити у користувача ввести ПІБ (*user_name*).
 - 7) Запросити у користувача ввести кваліфікацію.
 - 8) Запросити у користувача ввести бажаний розмір заробітної платні.
 - 9) Запросити у користувача ввести контактний телефон.
 - 10) Створити файл з іменем *user_url* *user_name.txt* (Наприклад, *Lab8_Petrenko.txt*)
 - 11) Записати у файл інформацію про претендента.
 - 12) Закрити файл.

- Зчитати/вивести на екран інформацію про потрібного претендента за введеним ім'ям:
 - 13) Запросити з екрану ім'я претендента.
 - 14) Вивести про нього інформацію.

- Робота з csv файлами:
- 15) Зчитати дані з файлу Data.csv
- 16) Вивести на екран перші 10 записів.

УВАГА! Слідкуйте за режимами доступу до файлу, своєчасно закривайте відкриті файли.

Індивідуальне завдання № 7. Визначити індекси елементів масиву (списку), значення яких належать заданому діапазону

Необхідно визначити індекси елементів списку, значення яких не менше заданого мінімуму і не більше заданого максимуму.

Нехай досліджуваний масив (список в Python) заповнюється випадковими числами в діапазоні від 0 до 99 (включно) і складається з 100 елементів.

Далі мінімум і максимум для пошуку значень задається користувачем.

Індивідуальне завдання № 8. Малювання з використанням черепашої графіки

Тема: Використання бібліотеки turtle, що дозволяє малювати на екрані нескладні малюнки.

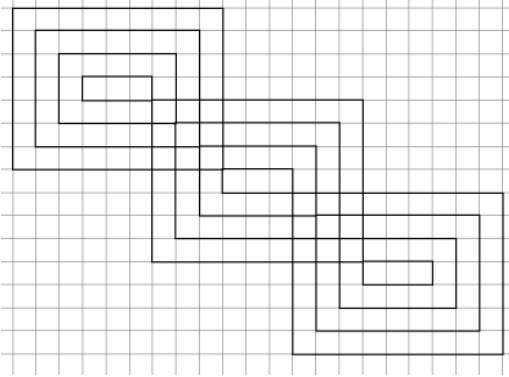
Обов'язкова наявність коментарів з умовами завдання перед написанням коду.

- 1) З використанням команд бібліотеки turtle намалювати багатокутник з N кутами, де N – число від 1 до 9 (остання цифра номеру за списком).
- 2) намалюйте на екрані рівносторонній трикутник. Намалюйте жовтий рівносторонній трикутник. Намалюйте зафарбований червоний рівносторонній трикутник.

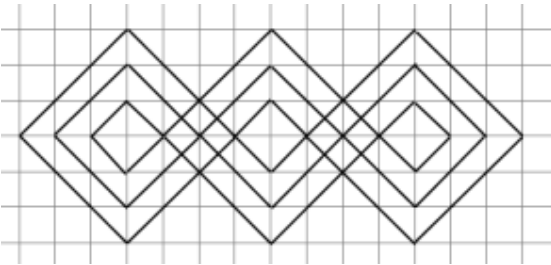
3) намалюйте на екрані квадрат з діагоналями. Намалюйте тільки діагоналі квадрата (два пересічних відрізка). Намалюйте квадрат, сторони якого не паралельні осям координат.

4) намалюйте:

Непарні номери за списком:



Парні номери за списком:



Всі програми, що працюють з черепашкою, повинні починатися з команд

```
import turtle
```

```
turtle.reset ()
```

а закінчуватися рядком

```
turtle._root.mainloop ()
```

Індивідуальне завдання № 9. Робота з списками в Python

Масиви в Пітоні називаються списками, тому що вони підтримують ряд додаткових операцій, що не властивих стандартним масивів. Заповніть список випадковими числами від 0 до N , де N – номер за списком у журналі ($N \geq 5$). Деякі елементи списку можна змінити.

```
import random
random.random ()
```

Потім:

- 1) Знайдіть найбільший елемент в списку.
- 2) Знайдіть найменший елемент в списку.
- 3) Знайдіть другий за величиною елемент у списку.
- 4) Знайдіть кількість елементів списку, рівних найбільшому.

Створити список студентів групи, відсортувати, додати ще одного студента.

Перетворити в кортеж.

Індивідуальне завдання № 10. Створення GUI додатку

Тема: Створити програмний додаток для перевірки валідності номеру кредитної/платіжної картки.

Обов'язкова наявність коментарів з умовами завдання перед написанням коду.

- 1) Тип проекту – Windows Form.

Користувач має ввести 16 цифр номеру картки, за натисканням кнопки «Check» має бути видано повідомлення або «The number 0123456789101112 is correct!!!» або «The number 0123456789101112 is NOT correct!!!». Приклад інтерфейсу програмного додатку наведено на рис. 1.

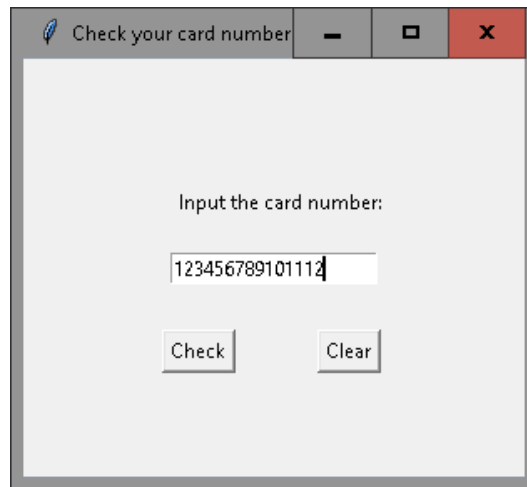


Рис. 1 Головна форма

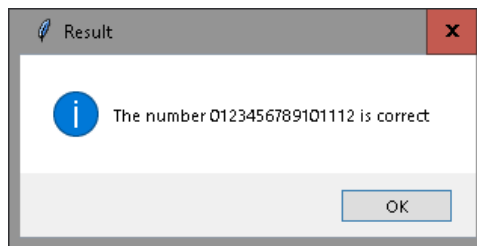


Рис. 2 Результат роботи програми

Перевірка введеного номера здійснюється за алгоритмом Луна (https://uk.wikipedia.org/wiki/%D0%90%D0%BB%D0%B3%D0%BE%D1%80%D0%B8%D1%82%D0%BC_%D0%9B%D1%83%D0%BD%D0%B0).

Також, визначити платіжну систему:

VISA – 4

American Express – 3

MasterCard – 5

Maestro - 3 або 6

JCB International - 3

China UnionPay – 6

<https://metanit.com/python/tutorial/9.2.php>

2) Тип проекту – Windows Form

Запуск виконання завдань здійснювати при натисканні на відповідні кнопки.

Уважно тестувати задачі.

Завдання для парних номерів у списку групи в проекті з іменем ElectroenergiyaВашеПрізвище

Завдання для непарних номерів у списку групи виконати у проекті з іменем ConverterВашеПрізвище,

Практична робота. Конвертер

Постановка задачі. Програма **Конвертер** перераховує ціну з доларів в гривні. Демонструє використання компонентів TextBox и Label для введення і відображення числових даних. Форму програми приведено на рис. 1

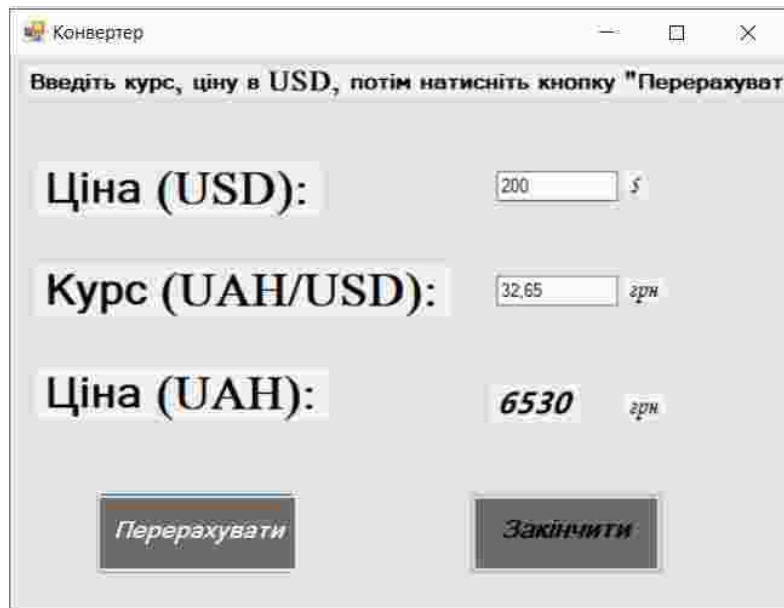


Рис.1. Інтерфейс додатку Конвертер

Програма **Електроенергія** визначає суму, яку потрібно заплатити користування електроенергією виходячи з показань лічильника. Форму програми наведено на рис. 2

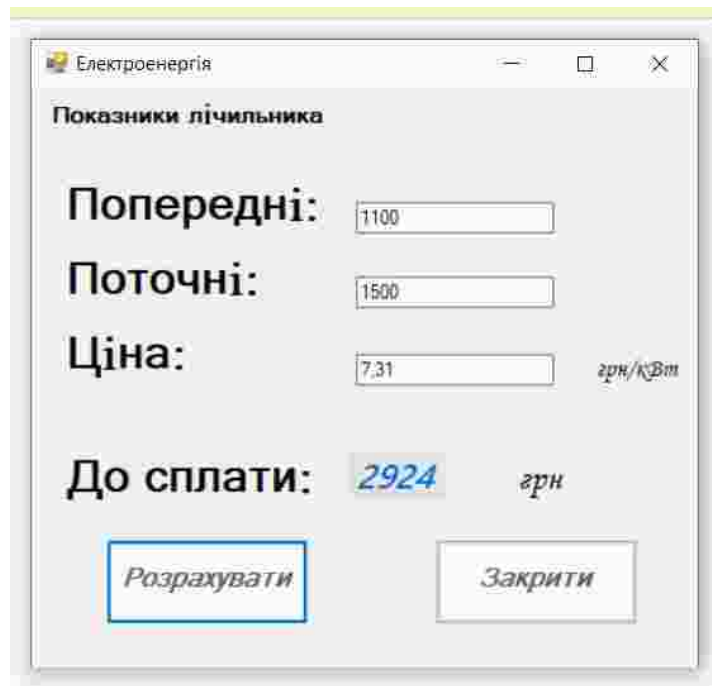


Рис.2. Інтерфейс додатку Електроенергія

Індивідуальне завдання № 11. Знайдення оптимального рішення

І застосування можливостей Pyton вирішити наступну задачу.

Підприємство випускає три види продукції А, Б і С (табл. 5.1) Для виробництва цієї продукції потрібні такі ресурси, як матеріали, праця робочих та ІТР. Для прийняття рішення оптимального випуску продукції, треба визначити такий план, який максимізує прибуток.

Дані студенти обирають згідно номеру за списком групи з табл. 1.

Таблиця 1

Вхідні данні

Варіант	Види витрат	Продукція			Обмеження за виробничими потужностями
		А	Б	С	
1	Витрати праці робочих на 1 тис. шт., годин	1	2	1	100
	Витрати матеріалів на 1 тис. шт., годин	1	0,4	0,45	75
	Витрати праці ІТР на 1 тис. шт., годин	2	2	7	295
	Прибуток на одну тисячу штук кожного виду продукції, тис. грн.	95	60	300	
2	Витрати праці робочих на 1 тис. шт., годин	2	1	1	100
	Витрати матеріалів на 1 тис. шт., годин	1	0,35	0,45	75
	Витрати праці ІТР на 1 тис. шт., годин	1	2	7	280
	Прибуток на одну тисячу штук кожного виду продукції, тис. грн.	95	65	300	
3	Витрати праці робочих на 1 тис. шт., годин	2	1	2	120
	Витрати матеріалів на 1 тис. шт., годин	1	0,4	0,45	75
	Витрати праці ІТР на 1 тис. шт., годин	1	2	6	275
	Прибуток на одну тисячу штук кожного виду продукції, тис. грн.	90	65	290	

Варіант	Види витрат	Продукція			Обмеження за виробничими потужностями
		А	Б	С	
4	Витрати праці робочих на 1 тис. шт., годин	2	1	2	110
	Витрати матеріалів на 1 тис. шт., годин	1	0,4	0,4	70
	Витрати праці ІТР на 1 тис. шт., годин	1	2	5	260
	Прибуток на одну тисячу штук кожного виду продукції, тис. грн.	90	65	290	
5	Витрати праці робочих на 1 тис. шт., годин	2	1	2	100
	Витрати матеріалів на 1 тис. шт., годин	2	0,45	0,4	80
	Витрати праці ІТР на 1 тис. шт., годин	2	2	6	260
	Прибуток на одну тисячу штук кожного виду продукції, тис. грн.	90	65	270	
6	Витрати праці робочих на 1 тис. шт., годин	2	1	2	95
	Витрати матеріалів на 1 тис. шт., годин	2	2	0,5	85
	Витрати праці ІТР на 1 тис. шт., годин	2	2	7	260
	Прибуток на одну тисячу штук кожного виду продукції, тис. грн.	90	60	265	
7	Витрати праці робочих на 1 тис. шт., годин	1	1	2	95
	Витрати матеріалів на 1 тис. шт., годин	2	3	0,5	120
	Витрати праці ІТР на 1 тис. шт., годин	2	2	7	260
	Прибуток на одну тисячу штук кожного виду продукції, тис. грн.	90	110	265	
8	Витрати праці робочих на 1 тис. шт., годин	1	1	2	95
	Витрати матеріалів на 1 тис. шт., годин	2	3	0,5	120
	Витрати праці ІТР на 1 тис. шт., годин	3	3	7	310
	Прибуток на одну тисячу штук кожного виду продукції, тис. грн.	90	125	265	

Варіант	Види витрат	Продукція			Обмеження за виробничими потужностями
		А	Б	С	
9	Витрати праці робочих на 1 тис. шт., годин	1	2	2	105
	Витрати матеріалів на 1 тис. шт., годин	2	3	0,5	120
	Витрати праці ІТР на 1 тис. шт., годин	2	3	6	310
	Прибуток на одну тисячу штук кожного виду продукції, тис. грн.	190	125	75	
10	Витрати праці робочих на 1 тис. шт., годин	1	2	2	125
	Витрати матеріалів на 1 тис. шт., годин	2	1	1	130
	Витрати праці ІТР на 1 тис. шт., годин	2	1	5	300
	Прибуток на одну тисячу штук кожного виду продукції, тис. грн.	190	120	170	
11	Витрати праці робочих на 1 тис. шт., годин	3	2	2	120
	Витрати матеріалів на 1 тис. шт., годин	2	3	3	135
	Витрати праці ІТР на 1 тис. шт., годин	2	4	5	300
	Прибуток на одну тисячу штук кожного виду продукції, тис. грн.	185	125	170	
12	Витрати праці робочих на 1 тис. шт., годин	3	5	4	125
	Витрати матеріалів на 1 тис. шт., годин	4	3	3	130
	Витрати праці ІТР на 1 тис. шт., годин	2	4	5	190
	Прибуток на одну тисячу штук кожного виду продукції, тис. грн.	180	135	175	
13	Витрати праці робочих на 1 тис. шт., годин	3	5	4	130
	Витрати матеріалів на 1 тис. шт., годин	4	6	7	135
	Витрати праці ІТР на 1 тис. шт., годин	5	4	5	195
	Прибуток на одну тисячу штук кожного виду продукції, тис. грн.	190	140	180	

Варіант	Види витрат	Продукція			Обмеження за виробничими потужностями
		А	Б	С	
14	Витрати праці робочих на 1 тис. шт., годин	6	5	4	140
	Витрати матеріалів на 1 тис. шт., годин	4	6	7	155
	Витрати праці ІТР на 1 тис. шт., годин	5	7	3	205
	Прибуток на одну тисячу штук кожного виду продукції, тис. грн.	200	150	210	
15	Витрати праці робочих на 1 тис. шт., годин	6	4	3	145
	Витрати матеріалів на 1 тис. шт., годин	7	6	7	160
	Витрати праці ІТР на 1 тис. шт., годин	5	7	6	210
	Прибуток на одну тисячу штук кожного виду продукції, тис. грн.	205	155	215	
16	Витрати праці робочих на 1 тис. шт., годин	6	4	4	150
	Витрати матеріалів на 1 тис. шт., годин	7	3	5	165
	Витрати праці ІТР на 1 тис. шт., годин	4	7	6	215
	Прибуток на одну тисячу штук кожного виду продукції, тис. грн.	225	160	220	
17	Витрати праці робочих на 1 тис. шт., годин	3	4	7	155
	Витрати матеріалів на 1 тис. шт., годин	7	3	5	170
	Витрати праці ІТР на 1 тис. шт., годин	4	5	6	230
	Прибуток на одну тисячу штук кожного виду продукції, тис. грн.	215	165	210	
18	Витрати праці робочих на 1 тис. шт., годин	2	2	1	90
	Витрати матеріалів на 1 тис. шт., годин	2	3	3	115
	Витрати праці ІТР на 1 тис. шт., годин	1	1	2	300
	Прибуток на одну тисячу штук кожного виду продукції, тис. грн.	270	120	260	

Варіант	Види витрат	Продукція			Обмеження за виробничими потужностями
		А	Б	С	
19	Витрати праці робочих на 1 тис. шт., годин	2	2	2	100
	Витрати матеріалів на 1 тис. шт., годин	2	3	3	125
	Витрати праці ІТР на 1 тис. шт., годин	3	4	2	250
	Прибуток на одну тисячу штук кожного виду продукції, тис. грн.	260	180	270	
20	Витрати праці робочих на 1 тис. шт., годин	1	2	2	105
	Витрати матеріалів на 1 тис. шт., годин	2	3	1	135
	Витрати праці ІТР на 1 тис. шт., годин	3	1	2	265
	Прибуток на одну тисячу штук кожного виду продукції, тис. грн.	255	195	260	
21	Витрати праці робочих на 1 тис. шт., годин	3	2	2	110
	Витрати матеріалів на 1 тис. шт., годин	2	3	1	145
	Витрати праці ІТР на 1 тис. шт., годин	1	1	2	245
	Прибуток на одну тисячу штук кожного виду продукції, тис. грн.	245	275	255	
22	Витрати праці робочих на 1 тис. шт., годин	3	3	2	115
	Витрати матеріалів на 1 тис. шт., годин	1	2	3	135
	Витрати праці ІТР на 1 тис. шт., годин	2	1	1	180
	Прибуток на одну тисячу штук кожного виду продукції, тис. грн.	255	285	270	
23	Витрати праці робочих на 1 тис. шт., годин	2	1	2	120
	Витрати матеріалів на 1 тис. шт., годин	1	2	3	130
	Витрати праці ІТР на 1 тис. шт., годин	2	1	1	190
	Прибуток на одну тисячу штук кожного виду продукції, тис. грн.	245	265	275	

Варіант	Види витрат	Продукція			Обмеження за виробничими потужностями
		А	Б	С	
24	Витрати праці робочих на 1 тис. шт., годин	2	4	1	125
	Витрати матеріалів на 1 тис. шт., годин	4	2	2	135
	Витрати праці ІТР на 1 тис. шт., годин	2	1	3	195
	Прибуток на одну тисячу штук кожного виду продукції, тис. грн.	255	275	285	
25	Витрати праці робочих на 1 тис. шт., годин	2	4	4	135
	Витрати матеріалів на 1 тис. шт., годин	4	2	2	140
	Витрати праці ІТР на 1 тис. шт., годин	3	3	3	200
	Прибуток на одну тисячу штук кожного виду продукції, тис. грн.	265	280	285	
26	Витрати праці робочих на 1 тис. шт., годин	5	4	4	155
	Витрати матеріалів на 1 тис. шт., годин	4	5	6	165
	Витрати праці ІТР на 1 тис. шт., годин	3	3	3	210
	Прибуток на одну тисячу штук кожного виду продукції, тис. грн.	275	290	280	
27	Витрати праці робочих на 1 тис. шт., годин	2	4	4	165
	Витрати матеріалів на 1 тис. шт., годин	4	2	2	170
	Витрати праці ІТР на 1 тис. шт., годин	3	3	3	205
	Прибуток на одну тисячу штук кожного виду продукції, тис. грн.	255	260	245	
28	Витрати праці робочих на 1 тис. шт., годин	5	4	4	160
	Витрати матеріалів на 1 тис. шт., годин	4	4	5	175
	Витрати праці ІТР на 1 тис. шт., годин	2	3	3	190
	Прибуток на одну тисячу штук кожного виду продукції, тис. грн.	275	250	260	

Рекомендовано використати бібліотеку pulp у такому порядку, як на прикладі:

```
from pulp import # відкриття бібліотеки
import time
start = time.time() # Визначення початку вирішення задачі
problem = pulp.LpProblem('0',pulp.LpMaximize)
# визначення найменувань змінних
x11 = pulp.LpVariable("x11", lowBound=0)
x12 = pulp.LpVariable ("x12", lowBound=0)
x13 = pulp.LpVariable("x13", lowBound=0)
x21 = pulp.LpVariable("x21", lowBound=0)
x22 = pulp.LpVariable("x22", lowBound=0)
x23 = pulp.LpVariable("x23", lowBound=0)
x31 = pulp.LpVariable ("x31", lowBound=0)
x32 = pulp.LpVariable("x32", lowBound=0)
x33 = pulp.LpVariable("x33", lowBound=0)
# скрипт початку максимізації задачі
# цільова функція
problem += - (7*x11 + 3*x12 + 6*x13 + 4*x21 + 8*x22 + 2*x23 + 1*x31 +
5*x32 + 9*x33), '---> MIN'
# обмеження. Після коми – номери обмежень
problem +=x11 + x12 + x13 == 20, '1'
problem +=x21 + x22 + x23 == 45, '2'
problem +=x31 + x32 + x33 == 30, '3'
problem +=x11 + x21 + x31 <= 74, '4'
problem +=x12 + x22 + x32 <= 40, '5'
problem +=x13 + x23 + x33 <= 36, '6'

problem.solve() #
```

```
print ("Числові значення змінних")
for variable in problem.variables():
    print (variable.name, "=", variable.varValue)
print ("Оптимальна вартість перевезень:")
print (abs(value(problem.objective)))
stop = time.time() # визначення часу закінчення вирішення задачі
print ("Час на вирішення оптимізаційної задачі", stop - start)
```

Екзаменаційне завдання. Робота з комплексом Python в Інтернеті

Ця робота буде зарахована як іспит.

Із сайту Державного управління статистики <http://www.ukrstat.gov.ua/> за допомогою команд Python:

1. Вибрати статистичну інформацію згідно номеру студента за списком групи за якнайбільший період. Конкретну таблицю в групі даних обирати довільно.
2. Зберегти цю інформацію на диску у файлі Excel.
3. Розрахувати середнє, дисперсію та стандарт для кожної колонки таблиці.
4. Розрахувати кореляційну матрицю зв'язку усіх колонок таблиці.
5. Представити кореляційну матрицю у вигляді графіку з точками.

№ за списком групи	Статистична інформація
1.	Ринок праці
2.	Освіта
3.	Охорона здоров'я
4.	Доходи та умови життя
5.	Соціальний захист
6.	Населені пункти та житло
7.	Правосуддя та злочинність
8.	Культура
9.	Макроекономічна статистика
10.	Національні рахунки
11.	Економічна діяльність
12.	Діяльність підприємств

№ за списком групи	Статистична інформація
13.	Внутрішня торгівля
14.	Капітальні інвестиції
15.	Основні засоби
16.	Сільське, лісове та рибне господарство
17.	Енергетика
18.	Промисловість
19.	Будівництво
20.	Транспорт
21.	Туризм
22.	Реєстр статистичних одиниць
23.	Навколишнє середовище
24.	Державні фінанси, податки та публічний сектор
25.	Зовнішньоекономічна діяльність
26.	Ціни
27.	Наука, технології та інновації
28.	Інформаційне суспільство

Контрольні запитання

1. Що таке машинне навчання ?
2. Назвіть два підходи до машинного навчання.
3. Наведіть постановку задачі навчання за прецедентами.

4. Наведіть постановку задачі дедуктивного навчання.
5. Визначте місце науки про дані в середовищі інформаційних технологій.
6. Які задачі, вирішуються за допомогою машинного навчання ?
7. Наведіть особливості обробки та застосування *big data* в економічних дослідженнях.
8. Наведіть визначення *big data*.
9. Яким вимогам мають задовольняти системи для накопичення *big data* ?
10. Відмінність методів *big data* від бізнес-аналізу.
11. Відмінність *big data* від малих даних.
12. В чому полягає складність обробки *big data* ?
13. Основні властивості *big data*.
14. Назвіть основні мови програмування для обробки *big data*.
15. Яке програмне забезпечення використовують для збору, накопичення, обробки та візуалізації *big data*.
16. В чому полягає ключова концепція *noSQL* і *Hadoop* про обробці *big data* ?
17. Назвіть основні компоненти екосистеми *Hadoop*.
18. Що таке HDFS?
19. Поясніть схему роботи *MapReduce*.
20. Що таке *Kaggle*?
21. Наведіть основні принципи роботи з *Kaggle*.
22. Що таке цикли в програмі?
23. Чим тримісний вираз `if / else` у пакеті *Python* відрізняється від звичайної конструкції `if / else`?
24. Як працювати з бібліотеками у пакеті *Python*?
25. Чим множини відрізняються від словників у пакеті *Python*?
26. Яка бібліотека дозволяє робити статистичні обчислення у пакеті *Python*?
27. Як вибирати дані зі сторінок в Інтернеті, написаних у кодах XML і HTML за допомогою команд пакету *Python*?