

[Skip to main content](#)

Side panel

Enter your search query

- [Допомога викладачам](#)
- [Інструкції](#)
- [Відео-уроки](#)
- [Вебінари](#)
- [Шаблони курсів](#)
- 
- [Допомога студентам](#)
- [Інструкції з користування CMS](#)
- [Розклад занять на порталі](#)
- [Зовнішні ресурси](#)
- [Документація Moodle](#)
- [Спільнота Moodle](#)
- [YouTube канал Learn Moodle Basics 3.8](#)

0

## Notifications

You have no notifications

[See all](#)

0

- English (en).
- Українська (uk).



- [Косик Дарина](#) 
- [Dashboard](#)[View profile](#)[Grades](#)[Preferences](#)[Notifications](#)[Calendar](#)[Log out](#)

[Home](#)

## Dashboard

## Events

## My Courses

[Історія європейської цивілізації \(2.2...](#)  
[Математичний аналіз 2 \(2.20-21.ПKN20/Б\).](#)  
[Дискретна математика 2 \(2.20-21.ПKN20/Б\).](#)  
[Основи програмування \(2.20-21.ПKN20/Б\).](#)  
[Математичний аналіз \(1.20-21.ПKN20/Б-...](#)  
[Дискретна математика \(1.20-21.ПKN20/Б...](#)  
[Основи програмування \(1.20-21.ПKN20/Б...](#)  
[Історія європейської цивілізації 1 \(1...](#)  
[СЯ.20-21.2. Світ і я. Біоетика](#)  
[СЯ. 20-21.2. Світ і я. Україна: істор...](#)  
[СЯ.1.20-21. Вступ до університетських...](#)  
[Computer Science English CSB \(2.20-21\).](#)  
[CS English B \(1.20-21\).](#)  
[Інформаційний ресурс для студентів І...](#)

This course

## Sections

## Основи програмування

## Тема 0. Вступ

## Тема 1. IHTEPHET, IP, TCP, HTTP, HTML, CSS

## Тема 2. MVC, FLASK

[Тема 3. Об'єктно-орієнтоване програмування. Основні поняття](#)  
[Тема 4. Інкапсуляція. Успадкування. Множинне успадкування. Поліморфізм.](#)  
[Тема 5. Абстрактні класи. Керування атрибутами. Методи класу та статичні методи](#)  
[Тема 6. Винятки, обробка винятків, ієрархія винятків, створення власних винятків. Тестування та зневадження програм.](#)  
[Тема 7. Розроблення графічного інтерфейсу користувача та візуалізація в Python.](#)  
[Попередній іспит](#)  
[Тема 8. Структури даних. Масиви.](#)  
[Тема 9. Структури даних. Зв'язані структури.](#)  
[Тема 10. Структури даних. Стек. Черга.](#)  
[Тема 11. Структури даних. Дерева. Бінарне дерево пошуку.](#)  
[Тема 12. Структури даних. Графи](#)  
[Іспит](#)  
[Домашнє завдання](#)

[Participants](#)

[Grades](#)

[Assignments](#)

[Quizzes](#)

[Resources](#)

[Virtual programming labs](#)

## [Site administration](#)



- [Home](#)
- [Dashboard](#)
- [Events](#)
- [My Courses](#)
  - [Історія європейської цивілізації \(2.20-21.ПКС20/Б\)](#)
  - [Математичний аналіз 2 \(2.20-21.ПКС20/Б\)](#)
  - [Дискретна математика 2 \(2.20-21.ПКС20/Б\)](#)
  - [Основи програмування \(2.20-21.ПКС20/Б\)](#)
  - [Математичний аналіз \(1.20-21.ПКС20/Б-...\)](#)
  - [Дискретна математика \(1.20-21.ПКС20/Б-...\)](#)
  - [Основи програмування \(1.20-21.ПКС20/Б-...\)](#)
  - [Історія європейської цивілізації 1 \(1.20-21.ПКС20/Б-...\)](#)
  - [СЯ.20-21.2. Світ і я. Біоетика](#)
  - [СЯ. 20-21.2. Світ і я. Україна: історія](#)
  - [СЯ.1.20-21. Вступ до університетських курсів](#)
  - [Computer Science English CSB \(2.20-21\)](#)
  - [CS English B \(1.20-21\)](#)
  - [Інформаційний ресурс для студентів І курсу](#)
- [This course](#)
  - [Sections](#)
    - [Основи програмування](#)
    - [Тема 0. Вступ](#)
    - [Тема 1. ІНТЕРНЕТ. IP, TCP, HTTP, HTML, CSS](#)
    - [Тема 2. MVC, FLASK](#)
    - [Тема 3. Об'єктно-орієнтоване програмування. Основні поняття](#)
    - [Тема 4. Інкапсуляція. Успадкування. Множинне успадкування. Поліморфізм.](#)
    - [Тема 5. Абстрактні класи. Керування атрибутами. Методи класу та статичні методи](#)
    - [Тема 6. Винятки, обробка винятків, ієрархія винятків, створення власних винятків. Тестування та зневадження програм.](#)
    - [Тема 7. Розроблення графічного інтерфейсу користувача та візуалізація в Python.](#)
    - [Попередній іспит](#)

- [Тема 8. Структури даних. Масиви.](#)
- [Тема 9. Структури даних. Зв'язані структури.](#)
- [Тема 10. Структури даних. Стек. Черга.](#)
- [Тема 11. Структури даних. Дерева. Бінарне дерево пошуку.](#)
- [Тема 12. Структури даних. Графи](#)
- [Іспит](#)
- [Домашнє завдання](#)
- [Participants](#)
- [Grades](#)
- [Assignments](#)
- [Quizzes](#)
- [Resources](#)
- [Virtual programming labs](#)

- Hide blocks

- 
- 

1. My courses
2. [Основи програмування \(2.20-21.ПКН20/Б\)](#)
3. [Тема 3. Об'єктно-орієнтоване програмування. Основні поняття](#)
4. [1. Навчальний Тест №1](#)

**Started on** Wednesday, 24 February 2021, 19:26

**State** Finished

**Completed on** Wednesday, 24 February 2021, 19:41

**Time taken** 15 mins

**Marks** 4.00/9.00

**Grade** 4.44 out of 10.00 (44%)

## Question 1

Correct

Mark 1.00 out of 1.00

☐  Not flaggedFlag question

### Question text

Яка послідовність запитів буде виконуватися при наступному сеансі роботи браузера. Користувач зайшов на сторінку блогу, авторизувався, перейшов до першого запису на окремій сторінці, написав та надіслав коментар до запису та перейшов на запис на іншій сторінці.

Select one:

- ☐ a. GET, POST, POST, GET, GET
- ☐ b. POST, GET, POST, GET, POST
- ☐ c. GET, POST, GET, GET, POST, GET
- ☐ d. GET, POST, GET, POST, POST, GET
- ☒ e. GET, POST, GET, POST, GET

### Feedback

Ваша відповідь правильна

The correct answer is: GET, POST, GET, POST, GET

## Question 2

Incorrect

Mark 0.00 out of 1.00

☐  Not flaggedFlag question

### Question text

Виберіть варіант пояснення що таке HTML, XML, JSON, який найкраще підходить для галузі веб-технологій.

Select one:

- ☒ Формати файлів, які використовуються для передачі даних в мережі згідно HTTP підходу до архітектури мережевих протоколів.
- ☐ Формати файлів, які використовуються для передачі даних в мережі згідно REST підходу до архітектури мережевих протоколів.
- ☐ Формати файлів, які не використовуються для передачі даних в мережі.
- ☐ Формати файлів.
- ☐ Типи даних, які можуть бути записані у файлі.
- ☐

Формати файлів, які використовуються для передачі даних.

### Feedback

Ваша відповідь неправильна

The correct answer is: Формати файлів, які використовуються для передачі даних в мережі згідно REST підходу до архітектури мережевих протоколів.

## Question 3

Correct

Mark 1.00 out of 1.00

☐  Not flaggedFlag question

### Question text

Встановіть відповідність між призначенням та назвою програми командного рядка.

is a console application of Windows operating systems that displays all current TCP/IP network configuration values and can modify Dynamic Host Configuration Protocol (DHCP) and Domain Name System (DNS) settings.

Answer 1

ipconfig ▼

is a system administration utility in Unix-like operating systems for network interface configuration

Answer 2

ifconfig ▼

is a network administration command-line tool available for many computer operating systems for querying the Domain Name System (DNS) to obtain domain name or IP address mapping or for any other specific DNS record.

Answer 3

nslookup ▼

is a computer network diagnostic tool for displaying the route (path) and measuring transit delays of packets across an Internet Protocol (IP) network. Answer 4 traceroute ▼

is a computer network administration software utility used to test the reachability of a host on an Internet Protocol (IP) network. Answer 5 ping ▼

## Feedback

Ваша відповідь правильна

The correct answer is: is a console application of Windows operating systems that displays all current TCP/IP network configuration values and can modify Dynamic Host Configuration Protocol (DHCP) and Domain Name System (DNS) settings. → ipconfig,

is a system administration utility in Unix-like operating systems for network interface configuration → ifconfig, is a network administration command-line tool available for many computer operating systems for querying the Domain Name System (DNS) to obtain domain name or IP address mapping or for any other specific DNS record. → nslookup, is a computer network diagnostic tool for displaying the route (path) and measuring transit delays of packets across an Internet Protocol (IP) network. → traceroute, is a computer network administration software utility used to test the reachability of a host on an Internet Protocol (IP) network. → ping

## Question 4

Correct

Mark 1.00 out of 1.00

☐  Not flaggedFlag question

## Question text

До яких рівнів згідно з мережевою моделлю OSI належать вказані протоколи?

TCP Answer 1 транспортний рівень ▼

IP Answer 2 мережевий рівень ▼

HTTP Answer 3 прикладний рівень ▼

## Feedback


Ваша відповідь правильна

The correct answer is: TCP → транспортний рівень, IP → мережевий рівень, HTTP → прикладний рівень

## Question 5

Not answered

Marked out of 1.00

☐  Not flaggedFlag question

### Question text

Якщо IP адреса та маска задані як 192.168.1.11\16 то яка адреса цього вузла у мережі (підмережі)?

Answer:

### Feedback

The correct answer is: 0.0.1.11

### Question 6

Correct

Mark 1.00 out of 1.00

☐  Not flaggedFlag question

### Question text

Об'єктно орієнтоване проектування це процес знаходження і визначення необхідних об'єктів для розв'язання проблеми.

Select one:

- ☐ True  
☒ False

### Feedback

The correct answer is 'False'.

### Question 7

Incorrect

Mark 0.00 out of 1.00

☐  Not flaggedFlag question

### Question text

Який метод потрібно розробити щоб користувач отримав зручне для сприйняття представлення екземпляра класу?

Answer:

### Feedback

The correct answer is: \_\_str\_\_

### Question 8

Incorrect

Mark 0.00 out of 1.00

☐  Not flaggedFlag question

### Question text

Чи є клас об'єктом?

Select one:

- ☐ True  
☒ False

### Feedback

The correct answer is 'True'.

### Question 9

Incorrect

Mark 0.00 out of 1.00

☐  Not flaggedFlag question

### Question text

Який метод першим виконується при створенні об'єкту, якщо у класі відсутній метод `__init__`?

Select one or more:

- ☐ new  
☒ `__init__` батьківського класу  
☐ `__init__()`  
☐ `__init__` попереднього класу в ієрархії класів  
☒ init  
☒ `self.__init__()`  
☐ `obj.self.__init__()`

### Feedback

Ваша відповідь неправильна

The correct answer is: `__init__` попереднього класу в ієрархії класів

Save the state of the flags

[Finish review](#)

[Previous Activity](#)

[Презентація](#)

[Next Activity](#)

[Хід виконання лабораторної роботи](#)

Jump to...  

[Skip Quiz navigation](#)

### Quiz navigation

[Question 1 This page](#) [Question 2 This page](#) [Question 3 This page](#) [Question 4 This page](#) [Question 5 This page](#) [Question 6 This page](#) [Question 7 This page](#) [Question 8 This page](#) [Question 9 This page](#)  
[Show one page at a time](#)[Finish review](#)

[Contacts](#)

Messages selected: 1 ×

[Contacts 0](#)

☐ Settings

- [Contacts](#)
- [Requests 0](#)

No contacts  
No contact requests

Contact request sent

Personal space

Save draft messages, links, notes etc. to access later.

☐ Delete for me and for everyone else

Block

Unblock

Remove

Add

Delete

Delete

Send contact request

Accept and add to contacts

Decline

OK

Cancel





Starred ()

No starred conversations

Group ()

No group conversations

Private ()

No private conversations

**Contacts**

## Non-contacts

Load more

## Messages

Load more

No results

Search people and messages

## Privacy

You can restrict who can message you

Accept messages from:

- ☐ My contacts only
- ☐ My contacts and anyone in my courses

## Notification preferences

### General

☐ Use enter to send

Write a message...

Search

Delete selected messages

Send contact request

You have blocked this user.

Unblock user

You are unable to message this user

[See all](#)

Український католицький університет

[www.ucu.edu.ua](http://www.ucu.edu.ua)

Вул. Іл. Свенціцького, 17, м. Львів, 79011

Тел.: (38/032) 240-99-40

Центр навчальних та інноваційних технологій

[підтримка користувачів]

[www.ceit.ucu.edu.ua](http://www.ceit.ucu.edu.ua)

Ел.пошта: [schurko@ucu.edu.ua](mailto:schurko@ucu.edu.ua)

Відділ інформаційних технологій

[підтримка системи]

[www.portal.ucu.edu.ua](http://www.portal.ucu.edu.ua)

Ел. пошта: [helpdesk@ucu.edu.ua](mailto:helpdesk@ucu.edu.ua)

Усі права застережено, (с) 2015-2021. [Український Католицький Університет](#)

[Data retention summary](#)

[Get the mobile app](#)

[Skip to main content](#)

Side panel

Enter your search query

- [Допомога викладачам](#)
- [Інструкції](#)
- [Відео-уроки](#)
- [Вебінари](#)
- [Шаблони курсів](#)
- 
- [Допомога студентам](#)
- [Інструкції з користування CMS](#)
- [Розклад занять на порталі](#)
- [Зовнішні ресурси](#)
- [Документація Moodle](#)
- [Спільнота Moodle](#)
- [YouTube канал Learn Moodle Basics 3.8](#)

0

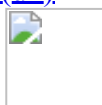
## Notifications

You have no notifications

[See all](#)

0

- - [English \(en\)](#)
  - [Українська \(uk\)](#)



- [Haronjuk Marta](#)

[Dashboard](#)[View profile](#)[Grades](#)[Preferences](#)[Notifications](#)[Calendar](#)[Log out](#)

[Home](#)

[Dashboard](#)

[Events](#)

[My Courses](#)

[Історія європейської цивілізації \(2.2...](#)  
[Математичний аналіз 2 \(2.20-21.ПКН20/Б\).](#)  
[Основи програмування \(2.20-21.ПКН20/Б\).](#)  
[Економічний аналіз 1 \(2.20-21.ПСА20/Б\).](#)  
[Математичний аналіз \(1.20-21.ПКН20/Б-...](#)  
[Дискретна математика \(1.20-21.ПКН20/Б...](#)  
[Основи програмування \(1.20-21.ПКН20/Б...](#)  
[Історія європейської цивілізації 1 \(1...](#)  
[СЯ. 20-21.2. Світ і я. Україна: істор...](#)  
[СЯ.20-21.2. Світ і я. Філософська ети...](#)  
[СЯ.1.20-21. Вступ до університетських...](#)  
[English for 1st year students \(IT BA\)...](#)

[This course](#)

[Sections](#)

[Основи програмування](#)

[Тема 0. Вступ](#)

[Тема 1. ІНТЕРНЕТ. IP, TCP, HTTP, HTML, CSS](#)

[Тема 2. MVC, FLASK](#)

[Тема 3. Об'єктно-орієнтоване програмування. Основні поняття](#)

[Тема 4. Інкапсуляція. Успадкування. Множинне успадкування.](#)

[Тема 5. Поліморфізм. Абстрактні класи. Методи класу та статичні методи](#)  
[Тема 6. Винятки, обробка винятків, ієрархія винятків, створення власних винятків. Тестування та зневадження програм.](#)  
[Тема 7. Розроблення графічного інтерфейсу користувача та візуалізація в Python.](#)  
[Регулярні вирази.](#)  
[Попередній іспит](#)  
[Тема 8. Структури даних. Масиви.](#)  
[Тема 9. Структури даних. Зв'язані структури.](#)  
[Тема 10. Структури даних. Стек. Черга.](#)  
[Тема 11. Структури даних. Дерева. Бінарне дерево пошуку.](#)  
[Тема 12. Структури даних. Графи](#)  
[Іспит](#)  
[Групове домашнє завдання](#)

[Participants](#)

[Grades](#)

[Assignments](#)

[Quizzes](#)

[Resources](#)

[Virtual programming labs](#)

[Site administration](#)



- [Home](#)
- [Dashboard](#)
- [Events](#)
- [My Courses](#)
  - [Історія європейської цивілізації \(2.2...](#)
  - [Математичний аналіз 2 \(2.20-21.ПКН20/Б\).](#)
  - [Основи програмування \(2.20-21.ПКН20/Б\).](#)
  - [Економічний аналіз 1 \(2.20-21.ПСА20/Б\).](#)
  - [Математичний аналіз \(1.20-21.ПКН20/Б-...](#)
  - [Дискретна математика \(1.20-21.ПКН20/Б...](#)
  - [Основи програмування \(1.20-21.ПКН20/Б...](#)
  - [Історія європейської цивілізації 1 \(1...](#)
  - [СЯ. 20-21.2. Світ і я. Україна: істор...](#)
  - [СЯ.20-21.2. Світ і я. Філософська ети...](#)
  - [СЯ.1.20-21. Вступ до університетських...](#)
  - [English for 1st year students \(IT BA\)...](#)
- [This course](#)
  - [Sections](#)
    - [Основи програмування](#)
    - [Тема 0. Вступ](#)
    - [Тема 1. ІНТЕРНЕТ. IP, TCP, HTTP, HTML, CSS](#)
    - [Тема 2. MVC, FLASK](#)
    - [Тема 3. Об'єктно-орієнтоване програмування. Основні поняття](#)
    - [Тема 4. Інкапсуляція. Успадкування. Множинне успадкування.](#)
    - [Тема 5. Поліморфізм. Абстрактні класи. Методи класу та статичні методи](#)
    - [Тема 6. Винятки, обробка винятків, ієрархія винятків, створення власних винятків. Тестування та зневадження програм.](#)
    - [Тема 7. Розроблення графічного інтерфейсу користувача та візуалізація в Python.](#)  
[Регулярні вирази.](#)
    - [Попередній іспит](#)
    - [Тема 8. Структури даних. Масиви.](#)
    - [Тема 9. Структури даних. Зв'язані структури.](#)
    - [Тема 10. Структури даних. Стек. Черга.](#)
    - [Тема 11. Структури даних. Дерева. Бінарне дерево пошуку.](#)

- [Тема 12. Структури даних. Графи](#)
- [Іспит](#)
- [Групове домашнє завдання](#)
- [Participants](#)
- [Grades](#)
- [Assignments](#)
- [Quizzes](#)
- [Resources](#)
- [Virtual programming labs](#)

- Show blocks
- 
- 

1. My courses
2. [Основи програмування \(2.20-21.ПКН20/Б\)](#)
3. [Тема 3. Об'єктно-орієнтоване програмування. Основні поняття](#)
4. [1. Навчальний Тест №1](#)

**Started on** Thursday, 25 February 2021, 21:04

**State** Finished

**Completed on** Thursday, 25 February 2021, 21:16

**Time taken** 11 mins 30 secs


**Marks** 5.77/9.00

**Grade** 6.41 out of 10.00 (64%)

## Question 1

Correct

Mark 1.00 out of 1.00

☐  Not flaggedFlag question

### Question text

Яка послідовність запитів буде виконуватися при наступному сеансі роботи браузера. Користувач зайшов на сторінку блогу, авторизувався, перейшов до першого запису на окремій сторінці, написав та надіслав коментар до запису та перейшов на запис на іншій сторінці.

Select one:

- ☐ a. GET, POST, GET, GET, POST, GET
- ☐ b. GET, POST, POST, GET, GET
- ☐ c. GET, POST, GET, POST, POST, GET
- ☐ d. POST, GET, POST, GET, POST
- ☒ e. GET, POST, GET, POST, GET

### Feedback

Ваша відповідь правильна

The correct answer is: GET, POST, GET, POST, GET

## Question 2

Partially correct

Mark 0.50 out of 1.00

☐  Not flaggedFlag question

### Question text

Що може відбутися, якщо вказати наступні параметри при запуску web-сервера, який розроблений на Flask.

```
app.run(debug=False, host='0.0.0.0', port=80)
```

Select one or more:

- ☒ Сервер не запуститься, бо вказаний порт 80 використовує протокол HTTP
- ☐ Сервер запуститься, але потрібно буде при його використанні вказувати адресу 0.0.0.0 та порт 80.
- ☐ Сервер не запуститься, бо вказано адресу 0.0.0.0
- ☐ Сервер запуститься за умови, що порт 80 вільний.
- ☐ Сервер запуститься, і його адреса 0.0.0.0

### Feedback

Ваша відповідь частково правильна.

You have correctly selected 1.

The correct answers are: Сервер не запуститься, бо вказаний порт 80 використовує протокол HTTP, Сервер запуститься за умови, що порт 80 вільний.

### Question 3

Correct

Mark 1.00 out of 1.00

☐  Not flaggedFlag question

### Question text

До яких рівнів згідно з мережевою моделлю OSI належать вказані протоколи?

TCP    Answer 1

HTTP    Answer 2

IP    Answer 3

### Feedback

Ваша відповідь правильна

The correct answer is: TCP → транспортний рівень, HTTP → прикладний рівень, IP → мережевий рівень

### Question 4



Partially correct

Mark 0.60 out of 1.00

☐  Not flaggedFlag question

### Question text

Встановіть відповідність між призначенням та назвою програми командного рядка.

is a computer network administration software utility used to test the reachability of a host on an Internet Protocol (IP) network.

Answer 1

tracert

is a system administration utility in Unix-like operating systems for network interface configuration

Answer 2

ifconfig

is a network administration command-line tool available for many computer operating systems for querying the Domain Name System (DNS) to obtain domain name or IP address mapping or for any other specific DNS record.

Answer 3

nslookup

is a computer network diagnostic tool for displaying the route (path) and measuring transit delays of packets across an Internet Protocol (IP) network.

Answer 4

ping

is a console application of Windows operating systems that displays all current TCP/IP network configuration values and can modify Dynamic Host Configuration Protocol (DHCP) and Domain Name System (DNS) settings.

Answer 5

ipconfig

### Feedback

Ваша відповідь частково правильна.

You have correctly selected 3.

The correct answer is: is a computer network administration software utility used to test the reachability of a host on an Internet Protocol (IP) network. → ping,

is a system administration utility in Unix-like operating systems for network interface configuration →

ifconfig, is a network administration command-line tool available for many computer operating systems for querying the Domain Name System (DNS) to obtain domain name or IP address mapping or for any

other specific DNS record. → nslookup, is a computer network diagnostic tool for displaying the route

(path) and measuring transit delays of packets across an Internet Protocol (IP) network. → tracert, is a


console application of Windows operating systems that displays all current TCP/IP network configuration values and can modify Dynamic Host Configuration Protocol (DHCP) and Domain Name System (DNS)

settings. → ipconfig

### Question 5

Incorrect

Mark 0.00 out of 1.00

☐  Not flaggedFlag question

### Question text

Якщо IP адреса та маска задані як 192.168.1.11/16 то яка адреса цієї мережі (підмережі)?

Answer: 0.0.1.1

### Feedback

The correct answer is: 192.168.0.0

### Question 6

Correct

Mark 1.00 out of 1.00

☐  Not flaggedFlag question

#### Question text

Об'єктно-орієнтований підхід до розроблення програмного забезпечення ґрунтується на написанні програм об'єктно-орієнтованою мовою програмування.

Select one:

- ☐ True  
☒ False


### Feedback

The correct answer is 'False'.

### Question 7

Incorrect

Mark 0.00 out of 1.00

☐  Not flaggedFlag question

#### Question text

Що буде результатом виконання наступного фрагмента коду:

```
class Test:
    def __init__(self):
        self.variable = 'Old'
        self.change(self.variable)

    def change(self, var):
        var = 'New'

obj=Test()
print(obj.variable)
```

Select one:

- ☐ На екран нічого не буде виводитись
- ☐ 'Old'
- ☒ Помилка
- ☐ 'New'

### Feedback

Ваша відповідь неправильна  
The correct answer is: 'Old'

### Question 8

Partially correct

Mark 0.67 out of 1.00

☐  Not flaggedFlag question

### Question text

Виберіть правильні твердження про об'єкт.

Select one or more:

- ☐ a. Об'єкти це система ідей і понять, які визначають стиль написання комп'ютерних програм.
- ☒ b. Об'єкти однакового типу мають спільну поведінку.
- ☐ c. Об'єкт в Python використовується для представлення типу.
- ☒ d. Об'єкт – це те, що характеризується значенням (станом), типом (поведінкою) та індивідуальністю.
- ☐ e. Для кожного об'єкту виділена його власна частина пам'яті.

### Feedback

Ваша відповідь частково правильна.

You have correctly selected 2.

The correct answers are: Об'єкт – це те, що характеризується значенням (станом), типом (поведінкою) та індивідуальністю., Об'єкти однакового типу мають спільну поведінку., Для кожного об'єкту виділена його власна частина пам'яті.

### Question 9

Correct

Mark 1.00 out of 1.00

☐  Not flaggedFlag question

### Question text

Який метод першим виконується при створенні об'єкту, якщо у класі відсутній метод `__init__`?

Select one or more:

- ☐ `__init__` батьківського класу
- ☐ `self.__init__()`
- ☐ `obj.self.__init__()`
- ☐ `init`
- ☒ `__init__` попереднього класу в ієрархії класів

- ☐ new  
☐ \_\_init\_\_()

## Feedback

Ваша відповідь правильна

The correct answer is: \_\_init\_\_ попереднього класу в ієрархії класів

Save the state of the flags

[Finish review](#)

[Previous Activity](#)

[Презентація](#)

[Next Activity](#)

[Хід виконання лабораторної роботи](#)

Jump to...

[Skip Quiz navigation](#)

## Quiz navigation

[Question 1 This page](#) [Question 2 This page](#) [Question 3 This page](#) [Question 4 This page](#) [Question 5 This page](#) [Question 6 This page](#) [Question 7 This page](#) [Question 8 This page](#) [Question 9 This page](#)  
[Show one page at a time](#) [Finish review](#)

Contacts

Messages selected: 1

[Contacts 0](#)

☐  
Settings

- [Contacts](#)
- [Requests 0](#)

No contacts

No contact requests

Contact request sent

Personal space

Save draft messages, links, notes etc. to access later.

☐ Delete for me and for everyone else

Block

Unblock

Remove

Add

Delete

Delete

Send contact request

Accept and add to contacts

Decline

OK

Cancel





Starred ()

No starred conversations

Group ()

No group conversations

Private ()

No private conversations

## Contacts

### Non-contacts

Load more

## Messages

Load more

No results

Search people and messages

## Privacy

You can restrict who can message you

Accept messages from:

- ☐ My contacts only
- ☐ My contacts and anyone in my courses

## Notification preferences

### General

☐ Use enter to send

Write a message...

Search

Delete selected messages

Send contact request

You have blocked this user.

Unblock user

You are unable to message this user

[See all](#)

Український католицький університет

[www.ucu.edu.ua](http://www.ucu.edu.ua)

Вул. Іл. Свенціцького, 17, м. Львів, 79011

Тел.: (38/032) 240-99-40

Центр навчальних та інноваційних технологій

[підтримка користувачів]

[www.ceit.ucu.edu.ua](http://www.ceit.ucu.edu.ua)

Ел.пошта: [schurko@ucu.edu.ua](mailto:schurko@ucu.edu.ua)

Відділ інформаційних технологій

[підтримка системи]



[www.portal.ucu.edu.ua](http://www.portal.ucu.edu.ua)

Ел. пошта: [helpdesk@ucu.edu.ua](mailto:helpdesk@ucu.edu.ua)

Усі права застережено, (с) 2015-2021. [Український Католицький Університет](#)

[Data retention summary](#)

[Get the mobile app](#)

[Skip to main content](#)

Side panel

Enter your search query

- [Допомога викладачам](#)
- [Інструкції](#)
- [Відео-уроки](#)
- [Вебінари](#)
- [Шаблони курсів](#)
- 
- [Допомога студентам](#)
- [Інструкції з користування CMS](#)
- [Розклад занять на порталі](#)
- [Зовнішні ресурси](#)
- [Документація Moodle](#)
- [Спільнота Moodle](#)
- [YouTube канал Learn Moodle Basics 3.8](#)

0

## Notifications

You have no notifications

[See all](#)

0

- English (en).
  - Українська (uk).



- [Косик Дарина](#)
- [Dashboard](#)[View profile](#)[Grades](#)[Preferences](#)[Notifications](#)[Calendar](#)[Log out](#)

[Home](#)

## Dashboard

## Events

## My Courses

[Історія європейської цивілізації \(2.2...](#)  
[Математичний аналіз 2 \(2.20-21.ПKN20/Б\).](#)  
[Дискретна математика 2 \(2.20-21.ПKN20/Б\).](#)  
[Основи програмування \(2.20-21.ПKN20/Б\).](#)  
[Математичний аналіз \(1.20-21.ПKN20/Б-...](#)  
[Дискретна математика \(1.20-21.ПKN20/Б...](#)  
[Основи програмування \(1.20-21.ПKN20/Б...](#)  
[Історія європейської цивілізації 1 \(1...](#)  
[СЯ.20-21.2. Світ і я. Біоетика](#)  
[СЯ. 20-21.2. Світ і я. Україна: істор...](#)  
[СЯ.1.20-21. Вступ до університетських...](#)  
[Computer Science English CSB \(2.20-21\).](#)  
[CS English B \(1.20-21\).](#)  
[Інформаційний ресурс для студентів І...](#)

This course

## Sections

## Основи програмування

## Тема 0. Вступ

## Тема 1. IHTEPHET, IP, TCP, HTTP, HTML, CSS

## Тема 2. MVC, FLASK

[Тема 3. Об'єктно-орієнтоване програмування. Основні поняття](#)  
[Тема 4. Інкапсуляція. Успадкування. Множинне успадкування.](#)  
[Тема 5. Поліморфізм. Абстрактні класи. Методи класу та статичні методи](#)  
[Тема 6. Винятки, обробка винятків, ієрархія винятків, створення власних винятків. Тестування та зневадження програм.](#)  
[Тема 7. Розроблення графічного інтерфейсу користувача та візуалізація в Python.](#)  
[Попередній іспит](#)  
[Тема 8. Структури даних. Масиви.](#)  
[Тема 9. Структури даних. Зв'язані структури.](#)  
[Тема 10. Структури даних. Стек. Черга.](#)  
[Тема 11. Структури даних. Дерева. Бінарне дерево пошуку.](#)  
[Тема 12. Структури даних. Графи](#)  
[Іспит](#)  
[Домашнє завдання](#)

[Participants](#)

[Grades](#)

[Assignments](#)

[Quizzes](#)

[Resources](#)

[Virtual programming labs](#)

[Site administration](#)



- [Home](#)
- [Dashboard](#)
- [Events](#)
- [My Courses](#)
  - [Історія європейської цивілізації \(2.2...](#)
  - [Математичний аналіз 2 \(2.20-21.ПKN20/Б\).](#)
  - [Дискретна математика 2 \(2.20-21.ПKN20/Б\).](#)
  - [Основи програмування \(2.20-21.ПKN20/Б\).](#)
  - [Математичний аналіз \(1.20-21.ПKN20/Б-...](#)
  - [Дискретна математика \(1.20-21.ПKN20/Б...](#)
  - [Основи програмування \(1.20-21.ПKN20/Б...](#)
  - [Історія європейської цивілізації 1 \(1...](#)
  - [СЯ.20-21.2. Світ і я. Біоетика](#)
  - [СЯ. 20-21.2. Світ і я. Україна: істор...](#)
  - [СЯ.1.20-21. Вступ до університетських...](#)
  - [Computer Science English CSB \(2.20-21\).](#)
  - [CS English B \(1.20-21\)](#)
  - [Інформаційний ресурс для студентів I...](#)
- [This course](#)
  - [Sections](#)
    - [Основи програмування](#)
    - [Тема 0. Вступ](#)
    - [Тема 1. ІНТЕРНЕТ. IP, TCP, HTTP, HTML, CSS](#)
    - [Тема 2. MVC, FLASK](#)
    - [Тема 3. Об'єктно-орієнтоване програмування. Основні поняття](#)
    - [Тема 4. Інкапсуляція. Успадкування. Множинне успадкування.](#)
    - [Тема 5. Поліморфізм. Абстрактні класи. Методи класу та статичні методи](#)
    - [Тема 6. Винятки, обробка винятків, ієрархія винятків, створення власних винятків. Тестування та зневадження програм.](#)
    - [Тема 7. Розроблення графічного інтерфейсу користувача та візуалізація в Python.](#)
    - [Попередній іспит](#)
    - [Тема 8. Структури даних. Масиви.](#)
    - [Тема 9. Структури даних. Зв'язані структури.](#)

- [Тема 10. Структури даних. Стек. Черга.](#)
- [Тема 11. Структури даних. Дерева. Бінарне дерево пошуку.](#)
- [Тема 12. Структури даних. Графи](#)
- [Іспит](#)
- [Домашнє завдання](#)
- [Participants](#)
- [Grades](#)
- [Assignments](#)
- [Quizzes](#)
- [Resources](#)
- [Virtual programming labs](#)

- Hide blocks
- 
- 

1. My courses
2. [Основи програмування \(2.20-21.ПKN20/Б\).](#)
3. [Тема 4. Інкапсуляція. Успадкування. Множинне успадкування.](#)
4. [2. Навчальний Тест №2](#)

**Started on** Thursday, 4 March 2021, 23:02

**State** Finished

**Completed on** Thursday, 4 March 2021, 23:12

**Time taken** 10 mins 1 sec

**Marks** 8.97/14.00

**Grade** 6.40 out of 10.00 (64%)

## Question 1

Correct

Mark 1.00 out of 1.00

☐  Not flaggedFlag question

### Question text

Яким буде результат виконання наступного фрагмента коду

```
class student:
    def __init__(self):
        self.marks = 97
        self.__cgpa = 8.7
    def display(self):
        print(self.marks)
obj=student()
print(obj._student__cgpa)
```

Answer:

### Feedback

The correct answer is: 8.7

## Question 2

Correct

Mark 1.00 out of 1.00

☐  Not flaggedFlag question

### Question text

Яким буде результат виконання наступного фрагмента коду

```
class Demo:
    def __init__(self):
        self.a = 1
        self.__b = 1

    def get(self):
        return self.__b

obj = Demo()
print(obj.get())
```

Answer:

### Feedback

The correct answer is: 1

## Question 3

Correct

Mark 1.00 out of 1.00

☐  Not flaggedFlag question

### Question text

Яким буде результат виконання наступного фрагмента коду

```
class Demo:
    def __init__(self):
        self.a = 11
        self.__b = 1

    def display(self):
        return self.__b

obj = Demo()
print(obj.a)
```

Answer:

### Feedback

The correct answer is: 11

### Question 4

Correct

Mark 1.00 out of 1.00

☐  Not flaggedFlag question

### Question text

Що буде виведено на екран після виконання наступного фрагмента коду?

```
class Fruits:
    def __init__(self, price):
        self.price = price

obj=Fruits(50)

obj.quantity=10
obj.bags=2

print(obj.quantity+len(obj.__dict__))
```

Answer:

### Feedback

The correct answer is: 13

### Question 5

Not answered

Marked out of 1.00

☐  Not flaggedFlag question

### Question text

Нехай А та В це екземпляри деякого класу С. Які методи та у якій послідовності будуть викликані при виконанні `print(A - B)`. **Вкажіть через кому і без додаткових пробілів.**

Answer:

### Feedback

The correct answer is: `__sub__()`, `__str__()`

## Question 6

Correct

Mark 1.00 out of 1.00

☐  Not flaggedFlag question

### Question text

Яка функція перевантажує оператор `>>` ?

Answer:

### Feedback

The correct answer is: `__rshift__`

## Question 7

Correct

Mark 1.00 out of 1.00

☐  Not flaggedFlag question

### Question text

Яка функція перевантажує оператор `//` ?

Answer:

### Feedback

The correct answer is: `__floordiv__`

## Question 8

Partially correct

Mark 0.17 out of 1.00

☐  Not flaggedFlag question

### Question text

Котрі з наступних тверджень правильні?

Select one or more:

- ☐ a. `super()` можна використовувати лише в першому рядку методу
- ☒ b. `super()` можна використовувати лише в методі `__init__()`
- ☒ c. Функція `super()` повертає екземпляр батьківського класу (об'єкт посередник, який дає доступ до методів попереднього класу в ієрархії класів)
- ☐ d. Виклик `super().__init__(self, x, y)` не спричинить помилку при виконанні
- ☐ e. У методі доцільно дублювати код з батьківського класу, якщо це можливо

### Feedback

Ваша відповідь частково правильна.  
You have correctly selected 1.

`super()` можна використовувати у всіх методах в будь-якому місці методу. Правильний виклик `super().__init__(x, y)`. `super()` справді повертає екземпляр батьківського класу (об'єкт посередник, який дає доступ до методів батьківського класу). Ніколи, ніколи не дублюйте код (якщо можете цього уникнути).

The correct answers are: Функція `super()` повертає екземпляр батьківського класу (об'єкт посередник, який дає доступ до методів попереднього класу в ієрархії класів), Виклик `super().__init__(self, x, y)` не спричинить помилку при виконанні

## Question 9

Partially correct

Mark 0.80 out of 1.00

☐  Not flaggedFlag question

### Question text

Виберіть правильні твердження про інкапсуляцію.

Select one or more:

- ☒ a. Інкапсуляція дозволяє захистити реалізацію методів від змін, що впливають на користувачів цього об'єкта.
- ☒ b. Інтерфейс до об'єкта реалізується через доступ до даних, збережених в об'єкті.
- ☒ c. Інкапсуляція (encapsulation) в мові програмування Python полягає в приховуванні деталей реалізації за інтерфейсом об'єкта.
- ☒ d. Деталі реалізації об'єкту визначені в класі, який ізолює решту програми від необхідності якось з ними стикатися.
- ☒ e. В об'єкті зберігаються дані та код, який використовує ці дані а їх взаємодія прихована.

### Feedback


Ваша відповідь частково правильна.  
You have selected too many options.

The correct answers are: Інкапсуляція (encapsulation) в мові програмування Python полягає в приховуванні деталей реалізації за інтерфейсом об'єкта., В об'єкті зберігаються дані та код, який використовує ці дані а їх взаємодія прихована., Деталі реалізації об'єкту визначені в класі, який ізолює решту програми від необхідності якось з ними стикатися.

## Question 10

Incorrect

Mark 0.00 out of 1.00

☐  Not flaggedFlag question

### Question text

Що буде виведено на екран в результаті виконання наступного фрагмента коду



```
>>> class A:
...     pass
>>> class B(A):
...     pass
>>> obj=B()
>>> isinstance(obj,A)
```

Answer:

### Feedback

The correct answer is: True

## Question 11

Not answered

Marked out of 1.00

☐  Not flaggedFlag question

### Question text

Що буде виведено на екран після виконання наступного фрагмента коду

```
>>> class A:
...     def test(self):
...         print(" test of A called ")
...
... class B(A):
...     def test(self):
...         print(" test of B called ")
...
... class C(B):
...     def test1(self):
...         print(" test of C called ")
...
... class D(C):
...     def test(self):
...         print(" test of D called ")
...         super().test()
... obj=D()
... obj.test()
```

  
[test of D called]  
[test of B called]

## Feedback

Ваша відповідь неправильна

## Question 12

Not answered

Marked out of 1.00

☐  Not flaggedFlag question

### Question text

Що буде виведено на екран в результаті виконання наступного фрагмента коду

```
class A:
    def __init__(self):
        self._x = 5
class B(A):
    def display(self):
        print(self._x)
def main():
    obj = B()
    obj.display()
main()
```

Answer:

## Feedback

The correct answer is: 5

## Question 13

Correct

Mark 1.00 out of 1.00

☐  Not flaggedFlag question

### Question text

Чи виконається безпомилково наступний код

```
class Test:
    def __init__(self):
        self.x = 0
class Derived_Test(Test):
    def __init__(self):
        self.y = 1
def main():
    b = Derived_Test()
    print(b.x,b.y)
main()
```

Select one:

- ☐ True  
☒ False

### Feedback

The correct answer is 'False'.

### Question 14

Correct

Mark 1.00 out of 1.00

☐  Not flaggedFlag question

### Question text

Який оператор перевантажується функцією \_\_or\_\_() ?

Answer: |

### Feedback

The correct answer is: |

Save the state of the flags

[Finish review](#)

[Previous Activity](#)

[Презентація](#)

[Next Activity](#)

[Хід виконання лабораторної роботи](#)

Jump to...  

[Skip Quiz navigation](#)

## Quiz navigation

[Question 1 This page](#) [Question 2 This page](#) [Question 3 This page](#) [Question 4 This page](#) [Question 5 This page](#) [Question 6 This page](#) [Question 7 This page](#) [Question 8 This page](#) [Question 9 This page](#) [Question 10 This page](#) [Question 11 This page](#) [Question 12 This page](#) [Question 13 This page](#) [Question 14 This page](#) [Show one page at a time](#) [Finish review](#)

Contacts

Messages selected: 1 

[Contacts 0](#)

  
Settings

- [Contacts](#)
- [Requests 0](#)

No contacts

No contact requests

Contact request sent

Personal space

Save draft messages, links, notes etc. to access later.

☐ Delete for me and for everyone else

Block

Unblock

Remove

Add

Delete

Delete

Send contact request

Accept and add to contacts

Decline

OK

Cancel



Starred ()

No starred conversations

Group ()

No group conversations

Private ()

No private conversations

## Contacts

### Non-contacts

Load more

## Messages

Load more

No results

Search people and messages

## Privacy

You can restrict who can message you

Accept messages from:

- ☐ My contacts only
- ☐ My contacts and anyone in my courses

## Notification preferences

### General

☐ Use enter to send

Write a message...

Search

Delete selected messages

Send contact request

You have blocked this user.

Unblock user

You are unable to message this user

[See all](#)

Український католицький університет

[www.ucu.edu.ua](http://www.ucu.edu.ua)

Вул. Іл. Свенціцького, 17, м. Львів, 79011

Тел.: (38/032) 240-99-40

Центр навчальних та інноваційних технологій

[підтримка користувачів]

[www.ceit.ucu.edu.ua](http://www.ceit.ucu.edu.ua)

Ел.пошта: [schurko@ucu.edu.ua](mailto:schurko@ucu.edu.ua)

Відділ інформаційних технологій

[підтримка системи]

[www.portal.ucu.edu.ua](http://www.portal.ucu.edu.ua)

Ел. пошта: [helpdesk@ucu.edu.ua](mailto:helpdesk@ucu.edu.ua)

Усі права застережено, (с) 2015-2021. [Український Католицький Університет](#)

[Data retention summary](#)

[Get the mobile app](#)



[Skip to main content](#)

Side panel

Enter your search query

- [Допомога викладачам](#)
- [Інструкції](#)
- [Відео-уроки](#)
- [Вебінари](#)
- [Шаблони курсів](#)
- 
- [Допомога студентам](#)
- [Інструкції з користування CMS](#)
- [Розклад занять на порталі](#)
- [Зовнішні ресурси](#)
- [Документація Moodle](#)
- [Спільнота Moodle](#)
- [YouTube канал Learn Moodle Basics 3.8](#)

0

## Notifications

You have no notifications

[See all](#)

0

- - [English \(en\)](#)
  - [Українська \(uk\)](#)



- [Косик Дарина](#)

[Dashboard](#)[View profile](#)[Grades](#)[Preferences](#)[Notifications](#)[Calendar](#)[Log out](#)

[Home](#)

[Dashboard](#)

[Events](#)

[My Courses](#)

[Історія європейської цивілізації \(2.20-21.ПКН20/Б\)](#)  
[Математичний аналіз 2 \(2.20-21.ПКН20/Б\)](#)  
[Дискретна математика 2 \(2.20-21.ПКН20/Б\)](#)  
[Основи програмування \(2.20-21.ПКН20/Б\)](#)  
[Математичний аналіз \(1.20-21.ПКН20/Б-...\)](#)  
[Дискретна математика \(1.20-21.ПКН20/Б-...\)](#)  
[Основи програмування \(1.20-21.ПКН20/Б-...\)](#)  
[Історія європейської цивілізації 1 \(1.20-21.ПКН20/Б-...\)](#)  
[СЯ.20-21.2. Світ і я. Біоетика](#)  
[СЯ. 20-21.2. Світ і я. Україна: історія](#)  
[СЯ.1.20-21. Вступ до університетських...](#)  
[Computer Science English CSB \(2.20-21\)](#)  
[CS English B \(1.20-21\)](#)  
[Інформаційний ресурс для студентів I...](#)

[This course](#)

[Sections](#)

[Основи програмування](#)

[Тема 0. Вступ](#)

[Тема 1. ІНТЕРНЕТ. IP, TCP, HTTP, HTML, CSS](#)

[Тема 2. MVC, FLASK](#)

[Тема 3. Об'єктно-орієнтоване програмування. Основні поняття](#)  
[Тема 4. Інкапсуляція. Успадкування. Множинне успадкування.](#)  
[Тема 5. Поліморфізм. Абстрактні класи. Методи класу та статичні методи](#)  
[Тема 6. Винятки, обробка винятків, ієрархія винятків, створення власних винятків. Тестування та зневадження програм.](#)  
[Тема 7. Розроблення графічного інтерфейсу користувача та візуалізація в Python.](#)  
[Попередній іспит](#)  
[Тема 8. Структури даних. Масиви.](#)  
[Тема 9. Структури даних. Зв'язані структури.](#)  
[Тема 10. Структури даних. Стек. Черга.](#)  
[Тема 11. Структури даних. Дерева. Бінарне дерево пошуку.](#)  
[Тема 12. Структури даних. Графи](#)  
[Іспит](#)  
[Домашнє завдання](#)

[Participants](#)

[Grades](#)

[Assignments](#)

[Quizzes](#)

[Resources](#)

[Virtual programming labs](#)

[Site administration](#)



- [Home](#)
- [Dashboard](#)
- [Events](#)
- [My Courses](#)
  - [Історія європейської цивілізації \(2.2...](#)
  - [Математичний аналіз 2 \(2.20-21.ПKN20/Б\).](#)
  - [Дискретна математика 2 \(2.20-21.ПKN20/Б\).](#)
  - [Основи програмування \(2.20-21.ПKN20/Б\).](#)
  - [Математичний аналіз \(1.20-21.ПKN20/Б-...](#)
  - [Дискретна математика \(1.20-21.ПKN20/Б...](#)
  - [Основи програмування \(1.20-21.ПKN20/Б...](#)
  - [Історія європейської цивілізації 1 \(1...](#)
  - [СЯ.20-21.2. Світ і я. Біоетика](#)
  - [СЯ. 20-21.2. Світ і я. Україна: істор...](#)
  - [СЯ.1.20-21. Вступ до університетських...](#)
  - [Computer Science English CSB \(2.20-21\).](#)
  - [CS English B \(1.20-21\)](#)
  - [Інформаційний ресурс для студентів I...](#)
- [This course](#)
  - [Sections](#)
    - [Основи програмування](#)
    - [Тема 0. Вступ](#)
    - [Тема 1. ІНТЕРНЕТ. IP, TCP, HTTP, HTML, CSS](#)
    - [Тема 2. MVC, FLASK](#)
    - [Тема 3. Об'єктно-орієнтоване програмування. Основні поняття](#)
    - [Тема 4. Інкапсуляція. Успадкування. Множинне успадкування.](#)
    - [Тема 5. Поліморфізм. Абстрактні класи. Методи класу та статичні методи](#)
    - [Тема 6. Винятки, обробка винятків, ієрархія винятків, створення власних винятків. Тестування та зневадження програм.](#)
    - [Тема 7. Розроблення графічного інтерфейсу користувача та візуалізація в Python.](#)
    - [Попередній іспит](#)
    - [Тема 8. Структури даних. Масиви.](#)
    - [Тема 9. Структури даних. Зв'язані структури.](#)

- [Тема 10. Структури даних. Стек. Черга.](#)
- [Тема 11. Структури даних. Дерева. Бінарне дерево пошуку.](#)
- [Тема 12. Структури даних. Графи](#)
- [Іспит](#)
- [Домашнє завдання](#)
- [Participants](#)
- [Grades](#)
- [Assignments](#)
- [Quizzes](#)
- [Resources](#)
- [Virtual programming labs](#)

- Hide blocks
- 
- 

1. My courses
2. [Основи програмування \(2.20-21.ПКи20/Б\).](#)
3. [Тема 4. Інкапсуляція. Успадкування. Множинне успадкування.](#)
4. [2. Навчальний Тест №2](#)

**Started on** Thursday, 4 March 2021, 22:51

**State** Finished

**Completed on** Thursday, 4 March 2021, 23:01

**Time taken** 10 mins

**Marks** 6.50/14.00

**Grade** 4.64 out of 10.00 (46%)

## Question 1

Correct

Mark 1.00 out of 1.00

☐  Not flaggedFlag question

### Question text

Чи буде на екрані зображено 1 після виконання наступного фрагмента коду?

```
class Demo:
    def __init__(self):
        self.a = 1
        self.__b = 1

    def display(self):
        return self.__b

obj = Demo()
print(obj.__b)
```

Select one:

- ☐ True
- ☒ False

## Feedback

The correct answer is 'False'.

## Question 2

Incorrect

Mark 0.00 out of 1.00

☐  Not flaggedFlag question

### Question text

До яких атрибутів не можна досягнути безпосередньо після виконання наступного фрагмента коду?

```
class Demo:
    def __init__(self):
        __a__ = 1
        self.__b__ = 1
        self.__c__ = 1
        __d__ = 1
```

Select one:

- ☐ self.\_\_c\_\_
- ☐ \_\_d\_\_
- ☒ \_\_a\_\_
- ☐ self.\_\_b

## Feedback

Ваша відповідь неправильна

The correct answer is: self.\_\_b

## Question 3

Correct

Mark 1.00 out of 1.00

☐  Not flaggedFlag question

### Question text

Яким буде результат виконання наступного фрагмента коду

```
class Demo:
    def __init__(self):
        self.a = 1
        self.__b = 1
    def get(self):
        return self.__b

obj = Demo()
obj.a=45
print(obj.a)
```

Answer:

### Feedback

The correct answer is: 45

### Question 4

Correct

Mark 1.00 out of 1.00

☐  Not flaggedFlag question

### Question text

Що буде виведено на екран після виконання наступного фрагмента коду?

```
class Demo:
    def __init__(self):
        pass

    def test(self):
        print(__name__)

obj = Demo()
obj.test()
```

Answer:

### Feedback

The correct answer is: \_\_main\_\_

### Question 5

Correct

Mark 1.00 out of 1.00

☐  Not flaggedFlag question

### Question text

Яка функція перевантажує оператор == ?

Answer:

### Feedback

The correct answer is: \_\_eq\_\_

### Question 6

Incorrect

Mark 0.00 out of 1.00

☐  Not flaggedFlag question

### Question text

Який оператор перевантажується функцією `__invert__()` ?

Answer:

### Feedback

The correct answer is: ~

### Question 7

Not answered

Marked out of 1.00

☐  Not flaggedFlag question

### Question text

Яка функція перевантажує оператор + (плюс)?

Answer:

### Feedback

The correct answer is: `__add__`

### Question 8

Partially correct

Mark 0.50 out of 1.00

☐  Not flaggedFlag question

### Question text

Для чого можна використати успадкування від вбудованих типів?

Select one or more:

- ☐ a. Створити клас точно такий самий як базовий, але з іншою назвою
- ☐ b. Щоб додати до їх інтерфейсу нові методи
- ☒ c. Щоб створити методи з такою ж імплементацією (реалізацією), як і в базовому типі, але з іншими назвами
- ☒ d. Створити аналог базового класу, змінивши реалізацію методів так щоб вони використовували інші алгоритми

### Feedback

Ваша відповідь частково правильна.

You have correctly selected 2.

Всі відповіді правильні. Додавання нових методів - типове використання успадкування, можна також перевизначити чинні методи. Перейменувати методи чи клас також можна, це може бути

потрібно, якщо треба, щоб базовий клас виглядав схожим на класи якоїсь бібліотеки, написаної користувачем.

The correct answers are: Щоб додати до їх інтерфейсу нові методи, Щоб створити методи з такою ж імплементацією (реалізацією), як і в базовому типі, але з іншими назвами, Створити аналог базового класу, змінивши реалізацію методів так щоб вони використовували інші алгоритми, Створити клас точно такий самий як базовий, але з іншою назвою

## Question 9

Correct

Mark 1.00 out of 1.00

☐  Not flaggedFlag question

### Question text

Уникнути подвійного виклику методу `__init__` (або конструктора) базового класу при diamond problem неможливо.

Select one:

- ☐ True  
☒ False

### Feedback

Функція `super()` змінює порядок виклику і дозволяє розв'язати цю проблему.

The correct answer is 'False'.

## Question 10

Not answered

Marked out of 1.00

☐  Not flaggedFlag question

### Question text

Якщо клас `Second` є підкласом класу `First` то якому способу виклику методу `__init__()` класу `First` з класу `Second` потрібно віддати перевагу?

Answer:


### Feedback

The correct answer is: `super().__init__()`

## Question 11

Correct

Mark 1.00 out of 1.00

☐  Not flaggedFlag question

### Question text

Чи виконається безпомилково наступний код

```
class A:
    def __init__(self):
        self.__x = 1
class B(A):
    def display(self):
        print(self.__x)
def main():
    obj = B()
    obj.display()
main()
```

Select one:

- ☐ True  
☒ False

### Feedback

The correct answer is 'False'.

## Question 12

Not answered

Marked out of 1.00

☐  Not flaggedFlag question

### Question text

Що буде виведено на екран в результаті виконання наступного фрагмента коду

```
class A:
    def __init__(self, x= 1):
        self.x = x
class der(A):
    def __init__(self,y = 2):
        super().__init__()
        self.y = y
def main():
    obj = der()
    print(obj.x, obj.y)
main()
```

Answer:

### Feedback

The correct answer is: 1 2

## Question 13

Not answered



Marked out of 1.00

☐  Not flaggedFlag question

### Question text

Що буде виведено на екран після виконання наступного фрагмента коду

```
class A:
    def test(self):
        print("test of A called")
class B(A):
    def test(self):
        print("test of B called")
        super().test()
class C(A):
    def test(self):
        print("test of C called")
        super().test()
class D(B,C):
    def test2(self):
        print("test of D called")
obj=D()
obj.test()
```

	[test of B called]
	[test of C called]
	[test of A called]

### Feedback

Ваша відповідь неправильна

### Question 14

Incorrect

Mark 0.00 out of 1.00

☐  Not flaggedFlag question

### Question text

Який клас забезпечить виведення на екран True при виконанні наступного фрагмента коду?

```
a = foo(2)
b = foo(3)
print(a < b)
```

Select one:

☒

```
class foo:
    def __init__(self, x):
        self.x = x
    def __lt__(self, other):
        if self.x < other.x:
            return False
        else:
            return True
```

☐

```
class foo:
    def __init__(self, x):
        self.x = x
    def __less__(self, other):
        if self.x < other.x:
            return False
        else:
            return True
```

☐

```
class foo:
    def __init__(self, x):
        self.x = x
    def __lt__(self, other):
        if self.x < other.x:
            return True
        else:
            return False
```

☐

```
class foo:
    def __init__(self, x):
        self.x = x
    def __less__(self, other):
        if self.x > other.x:
            return False
        else:
            return True
```

## Feedback

Ваша відповідь неправильна

```
class foo:
    def __init__(self, x):
        self.x = x
    def __lt__(self, other):
        if self.x < other.x:
            return True
        else:
            return False
```

The correct answer is:

Save the state of the flags

[Finish review](#)

[Previous Activity](#)

[Презентація](#)

[Next Activity](#)

[Хід виконання лабораторної роботи](#)

Jump to...

[Skip Quiz navigation](#)

## Quiz navigation

[Question 1 This page](#) [Question 2 This page](#) [Question 3 This page](#) [Question 4 This page](#) [Question 5 This page](#) [Question 6 This page](#) [Question 7 This page](#) [Question 8 This page](#) [Question 9 This page](#) [Question 10 This page](#) [Question 11 This page](#) [Question 12 This page](#) [Question 13 This page](#) [Question 14 This page](#) [Show one page at a time](#) [Finish review](#)

Contacts

Messages selected: 1

[Contacts 0](#)

☐

Settings

- [Contacts](#)
- [Requests 0](#)

No contacts

No contact requests

Contact request sent

Personal space

Save draft messages, links, notes etc. to access later.

☐ Delete for me and for everyone else





No starred conversations

Group ()

No group conversations

Private ()

No private conversations

## Contacts

### Non-contacts

Load more

## Messages

Load more

No results

Search people and messages

## Privacy

You can restrict who can message you

Accept messages from:

- ☐ My contacts only
- ☐ My contacts and anyone in my courses

## Notification preferences

### General

☐ Use enter to send

Write a message...

Search

Delete selected messages

Send contact request

You have blocked this user.

Unblock user

You are unable to message this user

[See all](#)

Український католицький університет

[www.ucu.edu.ua](http://www.ucu.edu.ua)

Вул. Іл. Свенціцького, 17, м. Львів, 79011

Тел.: (38/032) 240-99-40

Центр навчальних та інноваційних технологій

[підтримка користувачів]

[www.ceit.ucu.edu.ua](http://www.ceit.ucu.edu.ua)

Ел.пошта: [schurko@ucu.edu.ua](mailto:schurko@ucu.edu.ua)

Відділ інформаційних технологій

[підтримка системи]

[www.portal.ucu.edu.ua](http://www.portal.ucu.edu.ua)

Ел. пошта: [helpdesk@ucu.edu.ua](mailto:helpdesk@ucu.edu.ua)

Усі права застережено, (с) 2015-2021. [Український Католицький Університет](#)

[Data retention summary](#)

[Get the mobile app](#)

[Skip to main content](#)

Side panel

Enter your search query

- [Допомога викладачам](#)
- [Інструкції](#)
- [Відео-уроки](#)
- [Вебінари](#)
- [Шаблони курсів](#)
- 
- [Допомога студентам](#)
- [Інструкції з користування CMS](#)
- [Розклад занять на порталі](#)
- [Зовнішні ресурси](#)
- [Документація Moodle](#)
- [Спільнота Moodle](#)
- [YouTube канал Learn Moodle Basics 3.8](#)

0

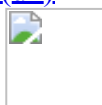
## Notifications

You have no notifications

[See all](#)

0

- - [English \(en\)](#)
  - [Українська \(uk\)](#)



- [Нагорнюк Марта](#)

[Dashboard](#)[View profile](#)[Grades](#)[Preferences](#)[Notifications](#)[Calendar](#)[Log out](#)

[Home](#)

[Dashboard](#)

[Events](#)

[My Courses](#)

[Історія європейської цивілізації \(2.2...](#)  
[Математичний аналіз 2 \(2.20-21.ПКН20/Б\)](#)  
[Основи програмування \(2.20-21.ПКН20/Б\)](#)  
[Економічний аналіз 1 \(2.20-21.ПСА20/Б\)](#)  
[Математичний аналіз \(1.20-21.ПКН20/Б-...](#)  
[Дискретна математика \(1.20-21.ПКН20/Б...](#)  
[Основи програмування \(1.20-21.ПКН20/Б...](#)  
[Історія європейської цивілізації 1 \(1...](#)  
[СЯ. 20-21.2. Світ і я. Україна: істор...](#)  
[СЯ.20-21.2. Світ і я. Філософська ети...](#)  
[СЯ.1.20-21. Вступ до університетських...](#)  
[English for 1st year students \(IT BA\)...](#)

[This course](#)

[Sections](#)

[Основи програмування](#)

[Тема 0. Вступ](#)

[Тема 1. ІНТЕРНЕТ. IP, TCP, HTTP, HTML, CSS](#)

[Тема 2. MVC, FLASK](#)

[Тема 3. Об'єктно-орієнтоване програмування. Основні поняття](#)

[Тема 4. Інкапсуляція. Успадкування. Множинне успадкування.](#)



[Тема 5. Поліморфізм. Абстрактні класи. Методи класу та статичні методи](#)  
[Тема 6. Винятки, обробка винятків, ієрархія винятків, створення власних винятків. Тестування та зневадження програм.](#)  
[Тема 7. Розроблення графічного інтерфейсу користувача та візуалізація в Python.](#)  
[Регулярні вирази.](#)  
[Попередній іспит](#)  
[Тема 8. Структури даних. Масиви.](#)  
[Тема 9. Структури даних. Зв'язані структури.](#)  
[Тема 10. Структури даних. Стек. Черга.](#)  
[Тема 11. Структури даних. Дерева. Бінарне дерево пошуку.](#)  
[Тема 12. Структури даних. Графи](#)  
[Іспит](#)  
[Групове домашнє завдання](#)

[Participants](#)

[Grades](#)

[Assignments](#)

[Quizzes](#)

[Resources](#)

[Virtual programming labs](#)

[Site administration](#)



- [Home](#)
- [Dashboard](#)
- [Events](#)
- [My Courses](#)
  - [Історія європейської цивілізації \(2.2...](#)
  - [Математичний аналіз 2 \(2.20-21.ПКН20/Б\).](#)
  - [Основи програмування \(2.20-21.ПКН20/Б\).](#)
  - [Економічний аналіз 1 \(2.20-21.ПСА20/Б\).](#)
  - [Математичний аналіз \(1.20-21.ПКН20/Б-...](#)
  - [Дискретна математика \(1.20-21.ПКН20/Б...](#)
  - [Основи програмування \(1.20-21.ПКН20/Б...](#)
  - [Історія європейської цивілізації 1 \(1...](#)
  - [СЯ. 20-21.2. Світ і я. Україна: істор...](#)
  - [СЯ.20-21.2. Світ і я. Філософська ети...](#)
  - [СЯ.1.20-21. Вступ до університетських...](#)
  - [English for 1st year students \(IT BA\)...](#)
- [This course](#)
  - [Sections](#)
    - [Основи програмування](#)
    - [Тема 0. Вступ](#)
    - [Тема 1. ІНТЕРНЕТ. IP, TCP, HTTP, HTML, CSS](#)
    - [Тема 2. MVC, FLASK](#)
    - [Тема 3. Об'єктно-орієнтоване програмування. Основні поняття](#)
    - [Тема 4. Інкапсуляція. Успадкування. Множинне успадкування.](#)
    - [Тема 5. Поліморфізм. Абстрактні класи. Методи класу та статичні методи](#)
    - [Тема 6. Винятки, обробка винятків, ієрархія винятків, створення власних винятків. Тестування та зневадження програм.](#)
    - [Тема 7. Розроблення графічного інтерфейсу користувача та візуалізація в Python.](#)  
[Регулярні вирази.](#)
    - [Попередній іспит](#)
    - [Тема 8. Структури даних. Масиви.](#)
    - [Тема 9. Структури даних. Зв'язані структури.](#)
    - [Тема 10. Структури даних. Стек. Черга.](#)
    - [Тема 11. Структури даних. Дерева. Бінарне дерево пошуку.](#)

- [Тема 12. Структури даних. Графи](#)
- [Іспит](#)
- [Групове домашнє завдання](#)
- [Participants](#)
- [Grades](#)
- [Assignments](#)
- [Quizzes](#)
- [Resources](#)
- [Virtual programming labs](#)

- Show blocks
- 
- 

1. My courses
2. [Основи програмування \(2.20-21.ПКН20/Б\)](#)
3. [Тема 4. Інкапсуляція. Успадкування. Множинне успадкування.](#)
4. [2. Навчальний Тест №2](#)

**Started on** Saturday, 20 March 2021, 10:14

**State** Finished

**Completed on** Saturday, 20 March 2021, 10:20

**Time taken** 5 mins 34 secs

**Marks** 5.92/14.00

**Grade** 4.23 out of 10.00 (42%)

## Question 1

Not answered

Marked out of 1.00

☐  Not flaggedFlag question

### Question text

Яким буде результат виконання наступного фрагмента коду

```
class fruits:
    def __init__(self):
        self.price = 100
        self.__bags = 5
    def display(self):
        print(self.__bags)
obj=fruits()
obj.display()
```

Answer:


### Feedback

The correct answer is: 5

## Question 2

Incorrect

Mark 0.00 out of 1.00

☐  Not flaggedFlag question

### Question text

Яким буде результат виконання наступного фрагмента коду

```
class student:
    def __init__(self):
        self.marks = 97
        self.__cgpa = 8.7
    def display(self):
        print(self.marks)
obj=student()
print(obj._student__cgpa)
```

Answer:

### Feedback

The correct answer is: 8.7

## Question 3

Correct

Mark 1.00 out of 1.00

☐  Not flaggedFlag question

### Question text

Яким буде результат виконання наступного фрагмента коду

```
class Demo:
    def __init__(self):
        self.a = 1
        self.__b = 1

    def get(self):
        return self.__b

obj = Demo()
print(obj.get())
```

Answer:

## Feedback

The correct answer is: 1

## Question 4

Incorrect

Mark 0.00 out of 1.00

☐  Not flaggedFlag question

### Question text

Що буде виведено на екран після виконання наступного фрагмента коду?

```
class Fruits:
    def __init__(self, price):
        self.price = price

obj=Fruits(50)

obj.quantity=10
obj.bags=2

print(obj.quantity+len(obj.__dict__))
```

Answer:

## Feedback

The correct answer is: 13

## Question 5

Correct

Mark 1.00 out of 1.00

☐  Not flaggedFlag question

### Question text

Яка функція перевантажує оператор == ?

Answer:

## Feedback

The correct answer is: \_\_eq\_\_

## Question 6

Correct

Mark 1.00 out of 1.00

☐  Not flaggedFlag question

### Question text

Який оператор перевантажується функцією `__invert__()` ?

Answer:

### Feedback

The correct answer is: ~

### Question 7

Correct

Mark 1.00 out of 1.00

☐  Not flaggedFlag question

### Question text

Яка функція перевантажує оператор `//` ?

Answer:

### Feedback

The correct answer is: \_\_floordiv\_\_

### Question 8

Partially correct

Mark 0.17 out of 1.00

☐  Not flaggedFlag question

### Question text

Котрі з наступних тверджень правильні?

Select one or more:

- ☐ a. `super()` можна використовувати лише в першому рядку методу
- ☒ b. Функція `super()` повертає екземпляр батьківського класу (об'єкт посередник, який дає доступ до методів попереднього класу в ієрархії класів)
- ☒ c. `super()` можна використовувати лише в методі `__init__()`
- ☐ d. У методі доцільно дублювати код з батьківського класу, якщо це можливо
- ☐ e. Виклик `super().__init__(self, x, y)` не спричинить помилку при виконанні

### Feedback

Ваша відповідь частково правильна.

You have correctly selected 1.

`super()` можна використовувати у всіх методах в будь-якому місці методу. Правильний виклик `super().__init__(x, y)`. `super()` справді повертає екземпляр батьківського класу (об'єкт посередник,

який дає доступ до методів батьківського класу). Ніколи, ніколи не дублюйте код (якщо можете цього уникнути).

The correct answers are: Функція `super()` повертає екземпляр батьківського класу (об'єкт посередник, який дає доступ до методів попереднього класу в ієрархії класів), Виклик `super().__init__(self, x, y)` не спричинить помилку при виконанні

## Question 9

Correct

Mark 1.00 out of 1.00

☐  Not flaggedFlag question

### Question text

Уникнути подвійного виклику методу `__init__` (або конструктора) базового класу при diamond problem неможливо.

Select one:

- ☐ True  
☒ False

### Feedback

Функція `super()` змінює порядок виклику і дозволяє розв'язати цю проблему.

The correct answer is 'False'.

## Question 10

Partially correct

Mark 0.75 out of 1.00

☐  Not flaggedFlag question

### Question text

Що буде виведено на екран після виконання наступного фрагмента коду

```

>>> class A:
...     def test(self):
...         print(" test of A called ")
...
... class B(A):
...     def test(self):
...         print(" test of B called ")
...
... class C(B):
...     def test1(self):
...         print(" test of C called ")
...
... class D(C):
...     def test(self):
...         print(" test of D called ")
...         super().test()
... obj=D()
... obj.test()

```

test of D called	
test of C called	[test of B called]

## Feedback

Ваша відповідь частково правильна.

## Question 11

Not answered

Marked out of 1.00

☐  Not flaggedFlag question

## Question text

Що буде виведено на екран в результаті виконання наступного фрагмента коду

```

class A:
    def __init__(self, x= 1):
        self.x = x
class der(A):
    def __init__(self,y = 2):
        super().__init__()
        self.y = y
def main():
    obj = der()
    print(obj.x, obj.y)
main()

```

Answer:

### Feedback

The correct answer is: 1 2

## Question 12

Incorrect

Mark 0.00 out of 1.00

☐  Not flaggedFlag question

### Question text

Що буде виведено на екран після виконання наступного фрагмента коду

```

class A:
    def test(self):
        print("test of A called")
class B(A):
    def test(self):
        print("test of B called")
        super().test()
class C(A):
    def test(self):
        print("test of C called")
        super().test()
class D(B,C):
    def test2(self):
        print("test of D called")
obj=D()
obj.test()

```



### Feedback

Ваша відповідь неправильна

### Question 13

Not answered

Marked out of 1.00

☐  Not flaggedFlag question

### Question text

Що буде виведено на екран в результаті виконання наступного фрагмента коду

```
class Test:
    def __init__(self):
        self.x = 0
class Derived_Test(Test):
    def __init__(self):
        Test.__init__(self)
        self.y = 1
def main():
    b = Derived_Test()
    print(b.x,b.y)
main()
```

Answer:

### Feedback

The correct answer is: 0 1

### Question 14

Not answered

Marked out of 1.00

☐  Not flaggedFlag question

### Question text

Яка функція перевантажує оператор + (плюс)?

Answer:

### Feedback

The correct answer is: add

Save the state of the flags

[Finish review](#)

[Previous Activity](#)

[Презентація](#)

[Next Activity](#)

[Хід виконання лабораторної роботи](#)

Jump to...

[Skip Quiz navigation](#)

## Quiz navigation

[Question 1 This page](#) [Question 2 This page](#) [Question 3 This page](#) [Question 4 This page](#) [Question 5 This page](#) [Question 6 This page](#) [Question 7 This page](#) [Question 8 This page](#) [Question 9 This page](#) [Question 10 This page](#) [Question 11 This page](#) [Question 12 This page](#) [Question 13 This page](#) [Question 14 This page](#) [Show one page at a time](#)[Finish review](#)

Contacts

Messages selected: 1

[Contacts 0](#)

☐ Settings

- [Contacts](#)
- [Requests 0](#)

No contacts

No contact requests

Contact request sent

Personal space

Save draft messages, links, notes etc. to access later.

☐ Delete for me and for everyone else





Starred ()

No starred conversations

Group ()

No group conversations

Private ()

No private conversations

## Contacts

### Non-contacts

Load more

## Messages

Load more

No results

Search people and messages

## Privacy

You can restrict who can message you

Accept messages from:

- ☐ My contacts only
- ☐ My contacts and anyone in my courses

## Notification preferences

### General

☐ Use enter to send

Write a message...

Search

Delete selected messages

Send contact request

You have blocked this user.

Unblock user

You are unable to message this user

[See all](#)

Український католицький університет

[www.ucu.edu.ua](http://www.ucu.edu.ua)

Вул. Іл. Свенціцького, 17, м. Львів, 79011

Тел.: (38/032) 240-99-40

Центр навчальних та інноваційних технологій

[підтримка користувачів]

[www.ceit.ucu.edu.ua](http://www.ceit.ucu.edu.ua)

Ел.пошта: [schurko@ucu.edu.ua](mailto:schurko@ucu.edu.ua)

Відділ інформаційних технологій

[підтримка системи]

[www.portal.ucu.edu.ua](http://www.portal.ucu.edu.ua)

Ел. пошта: [helpdesk@ucu.edu.ua](mailto:helpdesk@ucu.edu.ua)

Усі права застережено, (с) 2015-2021. [Український Католицький Університет](#)

[Data retention summary](#)

[Get the mobile app](#)



[Skip to main content](#)

Side panel

Enter your search query

- [Допомога викладачам](#)
- [Інструкції](#)
- [Відео-уроки](#)
- [Вебінари](#)
- [Шаблони курсів](#)
- 
- [Допомога студентам](#)
- [Інструкції з користування CMS](#)
- [Розклад занять на порталі](#)
- [Зовнішні ресурси](#)
- [Документація Moodle](#)
- [Спільнота Moodle](#)
- [YouTube канал Learn Moodle Basics 3.8](#)

1

## Notifications

You have no notifications

[See all](#)

0

- - [English \(en\)](#)
  - [Українська \(uk\)](#)



- [Косик Дарина](#)

[Dashboard](#)[View profile](#)[Grades](#)[Preferences](#)[Notifications](#)[Calendar](#)[Log out](#)

[Home](#)

[Dashboard](#)

[Events](#)

[My Courses](#)

[Історія європейської цивілізації \(2.20-21.ПКН20/Б\)](#)  
[Математичний аналіз 2 \(2.20-21.ПКН20/Б\)](#)  
[Дискретна математика 2 \(2.20-21.ПКН20/Б\)](#)  
[Основи програмування \(2.20-21.ПКН20/Б\)](#)  
[Математичний аналіз \(1.20-21.ПКН20/Б-...\)](#)  
[Дискретна математика \(1.20-21.ПКН20/Б-...\)](#)  
[Основи програмування \(1.20-21.ПКН20/Б-...\)](#)  
[Історія європейської цивілізації 1 \(1.20-21.ПКН20/Б-...\)](#)  
[СЯ.20-21.2. Світ і я. Біоетика](#)  
[СЯ. 20-21.2. Світ і я. Україна: історія](#)  
[СЯ.1.20-21. Вступ до університетських...](#)  
[Computer Science English CSB \(2.20-21\)](#)  
[CS English B \(1.20-21\)](#)  
[Інформаційний ресурс для студентів I...](#)

[This course](#)

[Sections](#)

[Основи програмування](#)

[Тема 0. Вступ](#)

[Тема 1. ІНТЕРНЕТ. IP, TCP, HTTP, HTML, CSS](#)

[Тема 2. MVC, FLASK](#)



[Тема 3. Об'єктно-орієнтоване програмування. Основні поняття](#)  
[Тема 4. Інкапсуляція. Успадкування. Множинне успадкування.](#)  
[Тема 5. Поліморфізм. Абстрактні класи. Методи класу та статичні методи](#)  
[Тема 6. Винятки, обробка винятків, ієрархія винятків, створення власних винятків. Тестування та зневадження програм.](#)  
[Тема 7. Розроблення графічного інтерфейсу користувача та візуалізація в Python.](#)  
[Попередній іспит](#)  
[Тема 8. Структури даних. Масиви.](#)  
[Тема 9. Структури даних. Зв'язані структури.](#)  
[Тема 10. Структури даних. Стек. Черга.](#)  
[Тема 11. Структури даних. Дерева. Бінарне дерево пошуку.](#)  
[Тема 12. Структури даних. Графи](#)  
[Іспит](#)  
[Групове домашнє завдання](#)

[Participants](#)

[Grades](#)

[Assignments](#)

[Quizzes](#)

[Resources](#)

[Virtual programming labs](#)

[Site administration](#)



- [Home](#)
- [Dashboard](#)
- [Events](#)
- [My Courses](#)
  - [Історія європейської цивілізації \(2.2...](#)
  - [Математичний аналіз 2 \(2.20-21.ПКН20/Б\).](#)
  - [Дискретна математика 2 \(2.20-21.ПКН20/Б\).](#)
  - [Основи програмування \(2.20-21.ПКН20/Б\).](#)
  - [Математичний аналіз \(1.20-21.ПКН20/Б-...](#)
  - [Дискретна математика \(1.20-21.ПКН20/Б...](#)
  - [Основи програмування \(1.20-21.ПКН20/Б...](#)
  - [Історія європейської цивілізації 1 \(1...](#)
  - [СЯ.20-21.2. Світ і я. Біоетика](#)
  - [СЯ. 20-21.2. Світ і я. Україна: істор...](#)
  - [СЯ.1.20-21. Вступ до університетських...](#)
  - [Computer Science English CSB \(2.20-21\).](#)
  - [CS English B \(1.20-21\)](#)
  - [Інформаційний ресурс для студентів I...](#)
- [This course](#)
  - [Sections](#)
    - [Основи програмування](#)
    - [Тема 0. Вступ](#)
    - [Тема 1. ІНТЕРНЕТ. IP, TCP, HTTP, HTML, CSS](#)
    - [Тема 2. MVC, FLASK](#)
    - [Тема 3. Об'єктно-орієнтоване програмування. Основні поняття](#)
    - [Тема 4. Інкапсуляція. Успадкування. Множинне успадкування.](#)
    - [Тема 5. Поліморфізм. Абстрактні класи. Методи класу та статичні методи](#)
    - [Тема 6. Винятки, обробка винятків, ієрархія винятків, створення власних винятків. Тестування та зневадження програм.](#)
    - [Тема 7. Розроблення графічного інтерфейсу користувача та візуалізація в Python.](#)
    - [Попередній іспит](#)
    - [Тема 8. Структури даних. Масиви.](#)
    - [Тема 9. Структури даних. Зв'язані структури.](#)

- [Тема 10. Структури даних. Стек. Черга.](#)
- [Тема 11. Структури даних. Дерева. Бінарне дерево пошуку.](#)
- [Тема 12. Структури даних. Графи](#)
- [Іспит](#)
- [Групове домашнє завдання](#)
- [Participants](#)
- [Grades](#)
- [Assignments](#)
- [Quizzes](#)
- [Resources](#)
- [Virtual programming labs](#)

- Hide blocks
- 
- 

1. My courses
2. [Основи програмування \(2.20-21.ПКН20/Б\).](#)
3. [Тема 5. Поліморфізм. Абстрактні класи. Методи класу та статичні методи](#)
4. [3. Навчальний Тест №3](#)

**Started on** Thursday, 18 March 2021, 10:36

**State** Finished

**Completed on** Thursday, 18 March 2021, 10:43


**Time taken** 7 mins

**Grade** **6.00** out of 10.00 (**60%**)

## Question 1

Correct

Mark 1.00 out of 1.00

☐  Not flaggedFlag question

### Question text

Об'єкти - екземпляри абстрактних класів завжди доцільно використовувати в об'єктно-орієнтованих програмах.

Select one:

- ☐ True
- ☒ False

### Feedback

The correct answer is 'False'.

## Question 2

Correct

Mark 1.00 out of 1.00

☐  Not flaggedFlag question

### Question text

В класі нащадку абстрактного класу не обов'язково розробляти реалізації всіх абстрактних методів, які є в батьківському класі.

Select one:

- ☒ True  
☐ False

### Feedback

The correct answer is 'True'.

### Question 3

Correct

Mark 1.00 out of 1.00

☐  Not flaggedFlag question

### Question text

Чи еквівалентні ці дві реалізації property:

#1

```
@property  
def silly(self):  
    return
```

---

#2

```
def set_silly(self):  
    return
```

```
silly = property(None, set_silly)
```

Select one:

- ☐ True  
☒ False

### Feedback

The correct answer is 'False'.

### Question 4

Correct

Mark 1.00 out of 1.00

☐  Not flaggedFlag question

### Question text

Якщо для створення property скористатися декоратором @property то результат буде такий самий що при використанні функції property().

Select one:

- ☐ True  
☒ False

### Feedback

The correct answer is 'False'.

## Question 5

Correct

Mark 1.00 out of 1.00

☐  Not flaggedFlag question

### Question text

Декоратор - це функція, яка приймає функцію або метод як аргумент, додає до неї додаткову логіку і повертає змінену функцію.

Select one:

- ☒ True  
☐ False

### Feedback

The correct answer is 'True'.

## Question 6

Correct

Mark 1.00 out of 1.00

☐  Not flaggedFlag question

### Question text

Абстрактний клас містить 7 методів. Скільки з цих методів повинні мати реалізацію у класі нащадку?

Select one or more:

- ☐ 6  
☐ 2  
☐ 7  
☐ 5  
☒ 1  
☐ 0

### Feedback

Ваша відповідь правильна  
The correct answer is: 1

## Question 7

Incorrect

Mark 0.00 out of 1.00

☐  Not flaggedFlag question

### Question text

Python дозволяє доступ до атрибутів класу за допомогою оператора `.` (dot notation). Такий спосіб простий, читабельний і короткий. Також використовують методи "приховування" атрибутів, до яких обмежується доступ через крапку, але також можна створювати властивості (property) щоб до атрибутів таки можна було досягнути через крапку. Які переваги такого підходу?

Select one or more:

- ☒ Це дозволяє спростити та скоротити обсяг коду.
- ☒ Від інших програмістів буде приховано інтерфейс.
- ☐ Це дозволяє контролювати атрибути.
- ☒ Такий спосіб захищає дані та обмежує доступ до них.
- ☒ Це дозволяє використовувати декоратор `@property` та функцію `property()`

## Feedback

Ваша відповідь неправильна

The correct answers are: Такий спосіб захищає дані та обмежує доступ до них., Це дозволяє контролювати атрибути.

## Question 8

Incorrect

Mark 0.00 out of 1.00

☐  Not flaggedFlag question

## Question text

Що буде виведено на екран у результаті виконання наступного фрагмента коду:

```
class A:
    def one(self):
        return self.two()

    def two(self):
        return 'A'

class B(A):
    def two(self):
        return 'B'

obj1=A()
obj2=B()
print(obj1.two(),obj2.one())
```

Answer:


## Feedback

The correct answer is: A B

## Question 9

Not answered

Marked out of 1.00

☐  Not flaggedFlag question

### Question text

Що буде результатом виконання наступного фрагмента коду:

```
class Demo:
    def __init__(self):
        self.x = 1
    def change(self):
        self.x = 10

class Demo_derived(Demo):
    def change(self):
        self.x=self.x + 1
        return self.x

def main():
    obj = Demo_derived()
    print(obj.change())

main()
```

Answer:

### Feedback

The correct answer is: 2

## Question 10

Not answered

Marked out of 1.00

☐  Not flaggedFlag question

### Question text

Що буде результатом виконання наступного фрагмента коду:

```

class A:
    def __init__(self,x=3):
        self._x = x

class B(A):
    def __init__(self):
        super().__init__(5)

    def display(self):
        print(self._x)

def main():
    obj = B()
    obj.display()

main()

```

Answer:

## Feedback

The correct answer is: 5

Save the state of the flags

[Finish review](#)

[Previous Activity](#)

[Презентація](#)

[Next Activity](#)

[Хід виконання лабораторної роботи](#)

Jump to...

[Skip Quiz navigation](#)

## Quiz navigation

[Question 1 This page](#) [Question 2 This page](#) [Question 3 This page](#) [Question 4 This page](#) [Question 5 This page](#) [Question 6 This page](#) [Question 7 This page](#) [Question 8 This page](#) [Question 9 This page](#) [Question 10 This page](#)

[Show one page at a time](#) [Finish review](#)

Contacts

Messages selected: 1

[Contacts 0](#)

- [Contacts](#)
- [Requests 0](#)

No contacts

No contact requests

Contact request sent

Personal space

Save draft messages, links, notes etc. to access later.

☐ Delete for me and for everyone else

Block

Unblock

Remove

Add

Delete

Delete

Send contact request

Accept and add to contacts

Decline

OK

Cancel





Starred ()

No starred conversations

Group ()

No group conversations

Private ()

No private conversations

## Contacts

## Non-contacts

Load more

## Messages

Load more

No results

Search people and messages

Privacy

You can restrict who can message you

Accept messages from:

☐ My contacts only

☐ My contacts and anyone in my courses

Notification preferences

General

☐ Use enter to send

Write a message...

Search

Delete selected messages

Send contact request

You have blocked this user.

Unblock user

You are unable to message this user

[See all](#)

Український католицький університет

[www.ucu.edu.ua](http://www.ucu.edu.ua)

Вул. Іл. Свенціцького, 17, м. Львів, 79011

Тел.: (38/032) 240-99-40

Центр навчальних та інноваційних технологій

[підтримка користувачів]

[www.ceit.ucu.edu.ua](http://www.ceit.ucu.edu.ua)

Ел.пошта: [schurko@ucu.edu.ua](mailto:schurko@ucu.edu.ua)

Відділ інформаційних технологій

[підтримка системи]

[www.portal.ucu.edu.ua](http://www.portal.ucu.edu.ua)

Ел. пошта: [helpdesk@ucu.edu.ua](mailto:helpdesk@ucu.edu.ua)

Усі права застережено, (с) 2015-2021. [Український Католицький Університет](#)

[Data retention summary](#)

[Get the mobile app](#)



Completed on	Wednesday, 24 February 2021, 19:41
Time taken	15 mins 1 sec
Marks	5.00/9.00
Grade	5.56 out of 10.00 (56%)

### Question 1

Correct

Mark 1.00 out of 1.00

🚩 Flag question

Яка послідовність запитів буде виконуватися при наступному сеансі роботи браузера. Користувач зайшов на сторінку блогу, авторизувався, перейшов до першого запису на окремій сторінці, написав та надіслав коментар до запису та перейшов на запис на іншій сторінці.

Select one:

- ☐ a. GET, POST, GET, GET, POST, GET
- ☐ b. POST, GET, POST, GET, POST
- ☒ c. GET, POST, GET, POST, GET ✓
- ☐ d. GET, POST, POST, GET, GET
- ☐ e. GET, POST, GET, POST, POST, GET

Ваша відповідь правильна

The correct answer is: GET, POST, GET, POST, GET

### Question 2

Incorrect

Mark 0.00 out of 1.00

🚩 Flag question

Виберіть варіант пояснення що таке HTML, XML, JSON, який найкраще підходить для галузі веб-технологій.

Select one:

- ☐ Формати файлів, які не використовуються для передачі даних в мережі.
- ☒ Формати файлів, які використовуються для передачі даних в мережі згідно HTTP підходу до архітектури мережевих протоколів. ✗
- ☐ Формати файлів, які використовуються для передачі даних.
- ☐ Типи даних, які можуть бути записані у файлі.
- ☐ Формати файлів, які використовуються для передачі даних в мережі згідно REST підходу до архітектури мережевих протоколів.

☐ Формати файлів.

Ваша відповідь неправильна

The correct answer is: Формати файлів, які використовуються для передачі даних в мережі згідно REST підходу до архітектури мережевих протоколів.

### Question 3

Correct

Mark 1.00 out of 1.00

🚩 Flag question

До яких рівнів згідно з мережевою моделлю OSI належать вказані протоколи?

IP	мережевий рівень	✓
HTTP	прикладний рівень	✓
TCP	транспортний рівень	✓

Ваша відповідь правильна

The correct answer is: IP → мережевий рівень, HTTP → прикладний рівень, TCP → транспортний рівень

### Question 4

Correct

Mark 1.00 out of 1.00

🚩 Flag question

Встановіть відповідність між призначенням та назвою програми командного рядка.

is a computer network diagnostic tool for displaying the route (path) and measuring transit delays of packets across an Internet Protocol (IP) network.

tracert



is a network administration command-line tool available for many computer operating systems for querying the Domain Name System (DNS) to obtain domain name or IP address mapping or for any other specific DNS record.

nslookup



is a system administration utility in Unix-like operating systems for network interface configuration

ifconfig



is a console application of Windows operating systems that displays all current TCP/IP network configuration values and can modify Dynamic Host Configuration Protocol (DHCP) and Domain Name System (DNS) settings.

ipconfig



is a computer network administration software utility used to test the reachability of a host on an Internet Protocol (IP) network.

ping



Ваша відповідь правильна

The correct answer is: is a computer network diagnostic tool for displaying the route (path) and measuring transit delays of packets across an Internet Protocol (IP) network. → traceroute, is a network administration command-line tool available for many computer operating systems for querying the Domain Name System (DNS) to obtain domain name or IP address mapping or for any other specific DNS record. → nslookup, is a system administration utility in Unix-like operating systems for network interface configuration → ifconfig, is a console application of Windows operating systems that displays all current TCP/IP network configuration values and can modify Dynamic Host Configuration Protocol (DHCP) and Domain Name System (DNS) settings. → ipconfig, is a computer network administration software utility used to test the reachability of a host on an Internet Protocol (IP) network. → ping

### Question 5

Incorrect

Mark 0.00 out of 1.00

🚩 Flag question

Якщо IP адреса та маска задані як 192.168.1.11\16 то яка адреса цієї мережі (підмережі)?

Answer: 192.168.1.11



The correct answer is: 192.168.0.0

### Question 6

Incorrect

Mark 0.00 out of 1.00

🚩 Flag question

Об'єктно орієнтоване проектування це процес знаходження і визначення необхідних об'єктів для розв'язання проблеми.

Select one:

☒ True



☐ False

The correct answer is 'False'.

### Question 7

Correct

Mark 1.00 out of 1.00

🚩 Flag question

Який метод потрібно розробити щоб користувач отримав зручне для сприйняття представлення екземпляра класу?

Answer:



The correct answer is: `__str__`

### Question 8

Correct

Mark 1.00 out of 1.00


🚩 Flag question

Чи успішно виконається такий код?

```
class MyClass:
    def __init__(self , x):
        self.x = x
    def myMethod():
        print(x)
```

```
c = MyClass(1234)
c.myMethod()
```

Select one:

- ☐ True
- ☒ False 

The correct answer is 'False'.

### Question 9

Incorrect

Mark 0.00 out of 1.00

🚩 Flag question

Який метод першим виконується при створенні об'єкту якщо у класі



Яким методом першим виконується при створенні об'єкту, якщо у класі відсутній метод `__init__`?

Select one or more:

- ☒ `self.__init__()` ✖
- ☒ `obj.self.__init__()` ✖
- ☐ `init`
- ☒ `__init__` попереднього класу в ієрархії класів ✔
- ☒ `__init__` батьківського класу ✖
- ☒ `__init__()` ✖
- ☒ `new` ✖

Ваша відповідь неправильна

The correct answer is: `__init__` попереднього класу в ієрархії класів

[Finish review](#)



Completed on	Thursday, 4 March 2021, 23:12
Time taken	10 mins
Marks	8.30/14.00
Grade	5.93 out of 10.00 (59%)

### Question 1

Correct

Mark 1.00 out of 1.00

🚩 Flag question

Яким буде результат виконання наступного фрагмента коду

```
class Demo:
    def __init__(self):
        self.a = 1
        self.__b = 1
    def get(self):
        return self.__b
obj = Demo()
obj.a=45
print(obj.a)
```

Answer: 45



The correct answer is: 45

### Question 2

Correct

Mark 1.00 out of 1.00

🚩 Flag question

Яким буде результат виконання наступного фрагмента коду

```
class student:
    def __init__(self):
        self.marks = 97
        self.__cgpa = 8.7
    def display(self):
        print(self.marks)
obj=student()
print(obj._student__cgpa)
```

Answer: 8.7



The correct answer is: 8.7

### Question 3

Correct

Mark 1.00 out of 1.00

🚩 Flag question

Яким буде результат виконання наступного фрагмента коду

```
class Demo:
    def __init__(self):
        self.a = 11
        self.__b = 1

    def display(self):
        return self.__b
obj = Demo()
print(obj.a)
```

Answer:



The correct answer is: 11

### Question 4

Correct

Mark 1.00 out of 1.00

🚩 Flag question

Що буде виведено на екран після виконання наступного фрагмента коду?

```
class Demo:
    def __init__(self):
        pass

    def test(self):
        print(__name__)

obj = Demo()
obj.test()
```

Answer:



The correct answer is: \_\_main\_\_

### Question 5

Incorrect

Mark 0.00 out of 1.00

Flag question

Який клас забезпечить виведення на екран True при виконанні наступного фрагмента коду?

```
a = foo(2)
b = foo(3)
print(a < b)
```

Select one:

☐

```
class foo:
    def __init__(self, x):
        self.x = x
    def __less__(self, other):
        if self.x < other.x:
            return False
        else:
            return True
```

☐

```
class foo:
    def __init__(self, x):
        self.x = x
    def __lt__(self, other):
        if self.x < other.x:
            return True
        else:
            return False
```

☒

```
class foo:
    def __init__(self, x):
        self.x = x
    def __lt__(self, other):
        if self.x < other.x:
            return False
        else:
            return True
```

 ✖

☐

```
class foo:
    def __init__(self, x):
        self.x = x
    def __less__(self, other):
        if self.x > other.x:
            return False
        else:
            return True
```

Ваша відповідь неправильна

The correct answer is:

```
class foo:
    def __init__(self, x):
        self.x = x
    def __lt__(self, other):
        if self.x < other.x:
            return True
        else:
            return False
```

### Question 6

Correct

Mark 1.00 out of 1.00

🚩 Flag question

Яка функція перевантажує оператор // ?

Answer:



The correct answer is: `__floordiv__`

### Question 7

Correct

Mark 1.00 out of 1.00

🚩 Flag question

Який оператор перевантажується функцією `__invert__()` ?

Answer:



The correct answer is: `~`

### Question 8

Incorrect

Mark 0.00 out of 1.00

🚩 Flag question

Виберіть правильні твердження про змінну класу. Вважайте, що `variable` - змінна класу, а `MyClass` - клас.

Select one or more:

- ☒ a. Змінна класу є спільною для всіх екземплярів класу ✓
- ☒ b. Щоб присвоїти значення змінній класу треба використовувати `self.variable = variable` ✗
- ☒ c. Змінну класу використовують для збереження даних спільних для всіх екземплярів класу ✓
- ☒ d. Доступ до значення змінної класу можна отримати лише з використанням синтаксису `MyClass.variable`. ✗

Ваша відповідь неправильна

Змінна класу спільна для всіх екземплярів, тож її можна(і треба)

використовувати для збереження даних спільних для всіх екземплярів. Якщо йдеться про доступ до даних змінної, варіанти `MyClass.variable` і `self.variable` повністю ідентичні, щоправда, перший підкреслює використання змінної класу, тому є кращим з точки зору стилю. Рядок `self.variable = variable` не змінить значення змінної класу, а лише створить змінну екземпляра з такою ж назвою.

The correct answers are: Змінна класу є спільною для всіх екземплярів класу, Змінну класу використовують для збереження даних спільних для всіх екземплярів класу

### Question 9

Partially correct

Mark 0.80 out of 1.00

🚩 Flag question

Виберіть правильні твердження про інкапсуляцію.

Select one or more:

- ☒ a. Інкапсуляція дозволяє захистити реалізацію методів від змін, що впливають на користувачів цього об'єкта. ✖
- ☒ b. Інкапсуляція (encapsulation) в мові програмування Python полягає в приховуванні деталей реалізації за інтерфейсом об'єкта. ✔
- ☒ c. В об'єкті зберігаються дані та код, який використовує ці дані а їх взаємодія прихована. ✔
- ☒ d. Деталі реалізації об'єкту визначені в класі, який ізолює решту програми від необхідності якимось з ними стикатися. ✔
- ☒ e. Інтерфейс до об'єкта реалізується через доступ до даних, збережених в об'єкті. ✖

Ваша відповідь частково правильна.

You have selected too many options.

The correct answers are: Інкапсуляція (encapsulation) в мові програмування Python полягає в приховуванні деталей реалізації за інтерфейсом об'єкта., В об'єкті зберігаються дані та код, який використовує ці дані а їх взаємодія прихована., Деталі реалізації об'єкту визначені в класі, який ізолює решту програми від необхідності якимось з ними стикатися.

### Question 10

Partially correct

Mark 0.25 out of 1.00

🚩 Flag question

Що буде виведено на екран після виконання наступного фрагмента коду

```
>>> class A:
...     def test(self):
...         print(" test of A called ")
```

```

...
... class B(A):
...     def test(self):
...         print(" test of B called ")
...
... class C(B):
...     def test1(self):
...         print(" test of C called ")
...
... class D(C):
...     def test(self):
...         print(" test of D called ")
...         super().test()
... obj=D()
... obj.test()

```

test of D called ✓

test of C called ✗ [test of B called]

test of B called ✗

test of A called ✗

Ваша відповідь частково правильна.

## Question 11

Not answered

Marked out of 1.00

🚩 Flag question

Що буде виведено на екран в результаті виконання наступного фрагмента коду

```

class A:
    def __init__(self,x):
        self.x = x
    def count(self,x):
        self.x = self.x+1
class B(A):
    def __init__(self, y=0):
        A.__init__(self, 3)
        self.y = y
    def count(self):
        self.y += 1
def main():
    obj = B()
    obj.count()
    print(obj.x, obj.y)
main()

```

Answer:  ✗

The correct answer is: 3 1

### Question 12

Partially correct

Mark 0.25 out of 1.00

🚩 Flag question

Що буде виведено на екран після виконання наступного фрагмента коду

```
class A:
    def test(self):
        print("test of A called")
class B(A):
    def test(self):
        print("test of B called")
        super().test()
class C(A):
    def test(self):
        print("test of C called")
        super().test()
class D(B,C):
    def test2(self):
        print("test of D called")
obj=D()
obj.test()
```

test of D called ✖ [test of B called]  
test of A called ✖ [test of C called]  
✖ [test of A called]  
✓

Ваша відповідь частково правильна.

### Question 13

Correct

Mark 1.00 out of 1.00

🚩 Flag question

Чи виконається безпомилково наступний код

```
class A:
    def __init__(self):
        self.__x = 1
class B(A):
    def display(self):
        print(self.__x)
def main():
    obj = B()
    obj.display()
```



```
main()
```

Select one:

- ☐ True
- ☒ False ✓

The correct answer is 'False'.

#### Question 14

Not answered

Marked out of 1.00

🚩 Flag question

Який оператор перевантажується функцією `__or__()` ?

Answer:  ✖

The correct answer is: |

[Finish review](#)



Completed on	Thursday, 4 March 2021, 23:01
Time taken	10 mins 11 secs
Marks	4.00/14.00
Grade	2.86 out of 10.00 (29%)

### Question 1

Correct

Mark 1.00 out of 1.00

🚩 Flag question

Яким буде результат виконання наступного фрагмента коду

```
class Demo:
    def __init__(self):
        self.a = 1
        self.__b = 1

    def get(self):
        return self.__b

obj = Demo()
print(obj.get())
```

Answer:



The correct answer is: 1

### Question 2

Correct

Mark 1.00 out of 1.00

🚩 Flag question

Яким буде результат виконання наступного фрагмента коду

```
class fruits:
    def __init__(self):
        self.price = 100
        self.__bags = 5
    def display(self):
        print(self.__bags)

obj=fruits()
obj.display()
```

Answer:



The correct answer is: 5

### Question 3

Correct

Mark 1.00 out of 1.00

🚩 Flag question

Чи буде на екрані зображено 1 після виконання наступного фрагмента коду?

```
class Demo:
    def __init__(self):
        self.a = 1
        self.__b = 1

    def display(self):
        return self.__b

obj = Demo()
print(obj.__b)
```

Select one:

- ☐ True
- ☒ False ✓

The correct answer is 'False'.

### Question 4

Correct

Mark 1.00 out of 1.00

🚩 Flag question

Що буде виведено на екран після виконання наступного фрагмента коду?

```
class Fruits:
    def __init__(self, price):
        self.price = price

obj=Fruits(50)

obj.quantity=10
obj.bags=2

print(obj.quantity+len(obj.__dict__))
```

Answer: 13



The correct answer is: 13

The correct answer is: 13

### Question 5

Incorrect

Mark 0.00 out of 1.00

🚩 Flag question

Нехай A та B це екземпляри деякого класу C. Які методи та у якій послідовності будуть викликані при виконанні `print(A - B)`. **Вкажіть через кому і без додаткових пробілів.**

Answer:  ❌

The correct answer is: `__sub__()`,`__str__()`

### Question 6

Not answered

Marked out of 1.00

🚩 Flag question

Яка функція перевантажує оператор `>>` ?

Answer:  ❌

The correct answer is: `__rshift__`

### Question 7

Not answered

Marked out of 1.00

🚩 Flag question

Яка функція перевантажує оператор `+` (плюс)?

Answer:  ❌

The correct answer is: `__add__`

### Question 8

Not answered

Marked out of 1.00

Котрі з наступних тверджень правильні?

Select one or more:

- ☐ a. У методі доцільно дублювати код з батьківського класу, якщо це можливо
- ☐ b. `super()` можна використовувати лише в методі `__init__()`
- ☐ c. `super()` можна використовувати лише в першому рядку методу
- ☐ d. Виклик `super().__init__(self, x, y)` не спричинить помилку при виконанні
- ☐ e. Функція `super()` повертає екземпляр батьківського класу (об'єкт посередник, який дає доступ до методів попереднього класу в ієрархії класів)

Ваша відповідь неправильна

`super()` можна використовувати у всіх методах в будь-якому місці методу. Правильний виклик `super().__init__(x, y)`. `super()` справді повертає екземпляр батьківського класу (об'єкт посередник, який дає доступ до методів батьківського класу). Ніколи, ніколи не дублюйте код (якщо можете цього уникнути).

The correct answers are: Функція `super()` повертає екземпляр батьківського класу (об'єкт посередник, який дає доступ до методів попереднього класу в ієрархії класів), Виклик `super().__init__(self, x, y)` не спричинить помилку при виконанні

### Question 9

Not answered

Marked out of 1.00

Flag question

Уникнути подвійного виклику методу `__init__` (або конструктора) базового класу при diamond problem неможливо.

Select one:

- ☐ True
- ☐ False

Функція `super()` змінює порядок виклику і дозволяє розв'язати цю проблему.  
The correct answer is 'False'.

### Question 10

Not answered

Marked out of 1.00

Flag question

Чи виконається безпомилково наступний код

```
class Test:
    def __init__(self):
        self.x = 0
class Derived_Test(Test):
    def __init__(self):
        self.y = 1
def main():
    b = Derived_Test()
    print(b.x,b.y)
main()
```

Select one:

- ☐ True
- ☐ False

The correct answer is 'False'.

### Question 11

Not answered

Marked out of 1.00

🚩 Flag question

Що буде виведено на екран в результаті виконання наступного фрагмента коду

```
class A:
    def __init__(self, x= 1):
        self.x = x
class der(A):
    def __init__(self,y = 2):
        super().__init__()
        self.y = y
def main():
    obj = der()
    print(obj.x, obj.y)
main()
```

Answer:



The correct answer is: 1 2

### Question 12

Not answered

Marked out of 1.00

🚩 Flag question

Що буде виведено на екран в результаті виконання наступного фрагмента коду

```
class Test:
    def __init__(self):
        self.x = 0
class Derived_Test(Test):
    def __init__(self):
        Test.__init__(self)
        self.y = 1
def main():
    b = Derived_Test()
    print(b.x,b.y)
main()
```

Answer:



The correct answer is: 0 1

### Question 13

Not answered

Marked out of 1.00

🚩 Flag question

Що буде виведено на екран в результаті виконання наступного фрагмента коду

```
class A:
    def __init__(self):
        self._x = 5
class B(A):
    def display(self):
        print(self._x)
def main():
    obj = B()
    obj.display()
main()
```

Answer:



The correct answer is: 5

### Question 14

Not answered

Marked out of 1.00

🚩 Flag question

Яка функція перевантажує оператор == ?

Answer:



The correct answer is:  $\_\_\text{eq}\_\_\_$

[Finish review](#)





Completed on	Saturday, 20 March 2021, 10:11
Time taken	7 mins 1 sec
Grade	5.67 out of 10.00 (57%)

### Question 1

Incorrect

Mark 0.00 out of 1.00

🚩 Flag question

Клас в якому принаймні один метод немає реалізації або реалізація полягає у збудження певного типу винятку називається абстрактним класом.

Select one:

- ☐ True
- ☒ False ✖

The correct answer is 'True'.

### Question 2

Correct

Mark 1.00 out of 1.00

🚩 Flag question

В класі нащадку абстрактного класу не обов'язково розробляти реалізації всіх абстрактних методів, які є в батьківському класі.

Select one:

- ☒ True ✔
- ☐ False

The correct answer is 'True'.

### Question 3

Incorrect

Mark 0.00 out of 1.00

🚩 Flag question

Чи еквівалентні ці дві реалізації property:

#1

```
@property  
def silly(self):
```

```
return
```

---

```
#2
```

```
def set_silly(self):  
    return
```

```
silly = property(set_silly, None, None, "my silly")
```

Select one:

- ☒ True ❌
- ☐ False

The correct answer is 'False'.

#### Question 4

Correct

Mark 1.00 out of 1.00

🚩 Flag question

Чи еквівалентні ці дві реалізації property:

```
#1
```

```
@property  
def silly(self):  
    return
```

---

```
#2
```

```
def set_silly(self):  
    return
```

```
silly = property(None, set_silly)
```

Select one:

- ☐ True
- ☒ False ✔️

The correct answer is 'False'.

#### Question 5

Correct

Mark 1.00 out of 1.00

🚩 Flag question

Декоратор - це функція, яка приймає функцію або метод як аргумент, додає до неї додаткову логіку і повертає змінену функцію.

Select one:

- ☒ True ✔️

☐ False

The correct answer is 'True'.

### Question 6

Partially correct

Mark 0.50 out of 1.00

🚩 Flag question

Абстрактний клас містить 7 методів. Скільки з цих методів повинні мати реалізацію у класі нащадку?

Select one or more:

☒ 1 ✓

☐ 6

☐ 7

☒ 0 ✗

☐ 2

☐ 5

Ваша відповідь частково правильна.

You have selected too many options.

The correct answer is: 1

### Question 7

Partially correct

Mark 0.17 out of 1.00

🚩 Flag question

Python дозволяє доступ до атрибутів класу за допомогою оператора `.` (dot notation). Такий спосіб простий, читабельний і короткий. Також використовують методи "приховування" атрибутів, до яких обмежується доступ через крапку, але також можна створювати властивості (property) щоб до атрибутів таки можна було доступитися через крапку. Навіщо надається така можливість?

Select one or more:

☐ Це дозволяє використовувати декоратор `@property` та функцію `property()`

☒ Це дозволяє контролювати атрибути. ✓

☒ Іншим програмістам буде надано зручний інтерфейс. ✓

☒ Це дозволяє скоротити обсяг коду. ✗

☐ Такий спосіб захищає дані та обмежує доступ до них.

Ваша відповідь частково правильна.

You have correctly selected 2.

The correct answers are: Такий спосіб захищає дані та обмежує доступ до них., Іншим програмістам буде надано зручний інтерфейс., Це дозволяє контролювати атрибути.

### Question 8

Correct

Mark 1.00 out of 1.00

🚩 Flag question

Що буде результатом виконання наступного фрагмента коду:

```
class A:
    def __init__(self):
        self.multiply(15)
        print(self.i)

    def multiply(self, i):
        self.i = 4 * i

class B(A):
    def __init__(self):
        super().__init__()
    def multiply(self, i):
        self.i = 2 * i

obj = B()
```

Answer: 30



The correct answer is: 30

### Question 9

Correct

Mark 1.00 out of 1.00

🚩 Flag question

Що буде виведено на екран у результаті виконання наступного фрагмента коду:

```
class A:
    def one(self):
        return self.two()
```

```
def two(self):
    return 'A'

class B(A):
    def two(self):
        return 'B'

obj1=A()
obj2=B()
print(obj1.two(),obj2.one())
```

Answer: A B



The correct answer is: A B

### Question 10

Incorrect

Mark 0.00 out of 1.00

Flag question

Що буде результатом виконання наступного фрагмента коду:

```
class Demo:
    def __init__(self):
        self.a = 1
        self.__b = 1

    def display(self):
        return self.__b

obj = Demo()
print(obj.__b)
```

Select one:

- ☒ Код виконається й на екран буде виведено 1. ✗
- ☐ Помилка, оскільки в класі відсутній метод `__str__`
- ☐ Помилка, оскільки метод `display` повертає значення приватної змінної.
- ☐ Помилка, оскільки до приватної змінної не можна доступитися безпосередньо.

Ваша відповідь неправильна

The correct answer is: Помилка, оскільки до приватної змінної не можна доступитися безпосередньо.





Completed on	Wednesday, 17 February 2021, 17:26
Time taken	2 mins 31 secs
Grade	1.00 out of 1.00 (100%)

### Question 1

Correct

Mark 0.17 out of 0.17

🚩 Flag question

Які засоби Python дозволяють отримати/розібрати вміст html сторінки?

Select one or more:

- ☒ a. requests ✓
- ☒ b. beautifulsoup ✓
- ☐ c. urlrequest
- ☐ d. geopy

Your answer is correct.

The correct answers are: requests, beautifulsoup

### Question 2

Correct

Mark 0.16 out of 0.16

🚩 Flag question

В контексті даного тесту, що таке API?

Select one:

- ☐ a. Aircraft Platform Interface
- ☒ b. Application Programming Interface ✓
- ☐ c. Asset Priority Index
- ☐ d. Application Programmers Initiative

Your answer is correct.

The correct answer is: Application Programming Interface

### Question 3

Correct

Mark 0.17 out of 0.17

🚩 Flag question

Які функції з модуля `json` слід використовувати при читанні даних у форматі `json`?

Select one or more:

- ☒ a. load ✓
- ☐ b. reads
- ☐ c. dump
- ☐ d. dumps
- ☒ e. loads ✓
- ☐ f. read

Your answer is correct.

The correct answers are: load, loads

#### Question 4

Correct

Mark 0.17 out of 0.17

🚩 Flag question

Вкажіть відповідність між типами даних у `json` форматі та їхніми відповідниками серед типів даних Python

object	dict	✓
array	list	✓
null	None	✓
real number	float	✓

Your answer is correct.

The correct answer is: object → dict, array → list, null → None, real number → float

#### Question 5

Correct

Mark 0.16 out of 0.16

🚩 Flag question

Коли потрібно надсилати запит на доступ до API?

Select one:



- ☒ a. Якомога швидше, адже заявку можуть розглядати певний час ✓
- ☐ b. В день дедлайну

Your answer is correct.

The correct answer is: Якомога швидше, адже заявку можуть розглядати певний час

### Question 6

Correct

Mark 0.17 out of 0.17

🚩 Flag question

Вкажіть порядок дій в базовій схемі роботи з API

- ✓отримання базового URL
- ✓формування запиту згідно API
- ✓надсилання запиту
- ✓обробка отриманих результатів запиту

Your answer is correct.

[Finish review](#)



Completed on	Wednesday, 24 February 2021, 19:21
Time taken	3 mins 7 secs
Grade	1.00 out of 1.00 (100%)

### Question 1

Correct

Mark 0.20 out of 0.20

🚩 Flag question

Виберіть правильну відповідність до малюнків.

<input type="text"/>	Узагальнення	✓
<input type="text"/>	Агрегація	✓
<input type="text"/>	Асоціація	✓
<input type="text"/>	Композиція	✓

Your answer is correct.

The correct answer is:  → Узагальнення,  
 → Агрегація,  → Асоціація,  
 → Композиція

### Question 2

Correct

Mark 0.20 out of 0.20

🚩 Flag question

За допомогою якого графічного примітиву ілюструють співвідношення класів як "ціле-частина", де частини не можуть існувати без цілого?

Select one:

- ☐ Асоціація
- ☒ Композиція ✓
- ☐ Узагальнення
- ☐ Агрегація

Your answer is correct.

The correct answer is: Композиція

### Question 3

Correct

Correct

Mark 0.20 out of 0.20

🚩 Flag question

Що є унікальним екземпляром структури даних, який визначається його класом?

Select one:

- ☐ Змінна
- ☐ Атрибут
- ☐ Метод
- ☒ Об'єкт ✓

Your answer is correct.

The correct answer is: Об'єкт

#### Question 4

Correct

Mark 0.20 out of 0.20

🚩 Flag question

Кожен з об'єктів класу має спільну поведінку, однаковий набір атрибутів і спільні значення.

Select one:

- ☐ True
- ☒ False ✓

The correct answer is 'False'.

#### Question 5

Correct

Mark 0.20 out of 0.20

🚩 Flag question

Який елемент графічного примітиву ілюструє таке відношення між класами, де клас «отримує» всі атрибути і методи класу, нащадком якого він є?

Select one:

- ☐ Клас
- ☐ Агрегація
- ☒ Узагальнення ✓

Your answer is correct.

The correct answer is: Узагальнення

[Finish review](#)



Completed on	Thursday, 25 March 2021, 12:46
Time taken	5 mins 23 secs
Marks	8.55/9.00
Grade	9.50 out of 10.00 (95%)

### Question 1

Correct

Mark 1.00 out of 1.00

🚩 Flag question

Встановіть відповідність між регулярними виразами та рядками які їм відповідають.

<code>apythex</code>	<code>^[AEIOUaeiou]</code>	✓
<code>apythex</code>	<code>[AEIOUaeiou]</code>	✓
<code>apythex</code>	<code>[^AEIOUaeiou]</code>	✓

Ваша відповідь правильна

The correct answer is: `apythex` → `^[AEIOUaeiou]`, `apythex` → `[AEIOUaeiou]`, `apythex` → `[^AEIOUaeiou]`

### Question 2

Correct

Mark 1.00 out of 1.00

🚩 Flag question

Встановіть відповідність між символами регулярних виразів та регулярними виразами.

<code>\w</code>	<code>[a-zA-Z0-9_]</code>	✓
<code>\W</code>	<code>[^a-zA-Z0-9_]</code>	✓
<code>\d</code>	<code>[0-9]</code>	✓
<code>\S</code>	<code>[^\t\n\r\f\v]</code>	✓
<code>\s</code>	<code>[\t\n\r\f\v]</code>	✓
<code>\D</code>	<code>[^0-9]</code>	✓

Ваша відповідь правильна

The correct answer is: `\w` → `[a-zA-Z0-9_]`, `\W` → `[^a-zA-Z0-9_]`, `\d` → `[0-9]`, `\S`

The correct answer is: `\w` → `[a-zA-Z0-9_]`, `\W` → `[^a-zA-Z0-9_]`, `\d` → `[0-9]`, `\S` → `[^\t\n\r\f\v]`, `\s` → `[\t\n\r\f\v]`, `\D` → `[^0-9]`

### Question 3

Correct

Mark 1.00 out of 1.00

🚩 Flag question

Що таке grep?

Select one:

- ☒ Програма командного рядка в Unix системах ✓
- ☐ Пакунок модулів Python
- ☐ Модуль Python
- ☐ Бібліотека Python
- ☐ Утиліта Windows

Ваша відповідь правильна

The correct answer is: Програма командного рядка в Unix системах

### Question 4

Correct

Mark 1.00 out of 1.00

🚩 Flag question

Кожна мова програмування передбачає власний синтаксис регулярних виразів.

Select one:

- ☐ True
- ☒ False ✓

The correct answer is 'False'.

### Question 5

Correct

Mark 1.00 out of 1.00

🚩 Flag question

Чи можна за допомогою регулярних виразів знайти в тексті смайлики?

Select one:

☒ True ✓

☐ False

The correct answer is 'True'.

### Question 6

Partially correct

Mark 0.80 out of 1.00

🚩 Flag question

Що буде виведено на екран в результаті виконання наступного фрагмента програми?

```
import re
programming = ["PHP", "C", "Python", "Java", "C++", ]

pat = "^B|^P|^A$|^+$"

for lang in programming:
    if re.search(pat, lang):
        print("{} {}".format(lang, "FOUND"))
    else:
        print("{} {}".format(lang, "NOT FOUND"))
```

PHP FOUND ✓

C NOT FOUND ✓

Python FOUND ✓

Java NOT FOUND ✓

C++ FOUND ✗ [C++ FOUND]

Ваша відповідь частково правильна.

### Question 7

Correct

Mark 1.00 out of 1.00

🚩 Flag question

Яким буде результат виконання наступного фрагмента коду?

```
import re
text = 'The quickbrown fox jumps over the lazy dog.'.split()
print(' '.join([w for w in text if re.findall(r"\w{5}", w)]))
```

quickbrown jumps ✓

### Question 8

Correct

Mark 1.00 out of 1.00

🚩 Flag question

Елементи яких електронних адрес будуть виведені на екран в результаті виконання наступних рядків коду?

```
import re
emails = """page26@facebook.com
page33@google.com.ua
page33@google.com
page42@gmail7.list
page42@gmail.ua
page14@true.c

pattern = r'([A-Za-z0-9]+)@([A-Za-z]+\.\w{2,4})'
print(re.findall(pattern, emails))
```

Select one or more:

- ☒ a. page33@google.com ✓
- ☒ b. page42@gmail.ua ✓
- ☐ c. page42@gmail7.list
- ☐ d. page14@true.c
- ☒ e. page26@facebook.com ✓
- ☒ f. page33@google.com.ua ✓

Ваша відповідь правильна

The correct answers are: page26@facebook.com, page33@google.com.ua, page33@google.com, page42@gmail.ua

### Question 9

Partially correct

Mark 0.75 out of 1.00

🚩 Flag question

Які з наступних викликів функцій модуля re дозволяють зі списку *text* отримати список слів з трьох літер, які починаються на літеру *b* (без врахування регістру)?

```
text = """Betty bought a bit of butter,
But the butter was so bitter,
So she bought some better butter,
To make the bitter butter better.""".split()
```

Select one or more:

- ☐ a. [w for w in text if re.match(r'^B|^b)\w{2,3}', w)]
- ☒ b. [w for w in text if re.findall(r'(B|b)\w{2,3}', w) and len(w) == 3] ✓
- ☐ c. [w for w in text if re.fullmatch(r'^B|^b)\w{2,3}', w)]



- ☒ c. [w for w in text if re.fullmatch(r'^B|^b)\w{2,3}', w)] ✓
- ☐ d. [w for w in text if re.fullmatch(r'(B|b)\w{,3}', w)]
- ☐ e. [w for w in text if re.match(r'(B|b)\w{3}', w)]
- ☒ f. [w for w in text if re.findall(r'(B|b)\w{,3}', w) and len(w) == 3] ✓
- ☐ g. [w for w in text if re.findall(r"(B|b)\w{3}", w)]

Ваша відповідь частково правильна.

You have correctly selected 3.

The correct answers are: [w for w in text if re.fullmatch(r'^B|^b)\w{2,3}', w)], [w for w in text if re.fullmatch(r'(B|b)\w{,3}', w)], [w for w in text if re.findall(r'(B|b)\w{,3}', w) and len(w) == 3], [w for w in text if re.findall(r'(B|b)\w{2,3}', w) and len(w) == 3]

[Finish review](#)



**Completed on** Wednesday, 10 February 2021, 10:35

**Time taken** 2 mins 54 secs

### Question 1

Complete

Not graded

🚩 Flag question

За допомогою яких елементів фізично реалізоване підключення до інтернету? Які **елементи** є складовими інтернет-з'єднання

Select one:

- ☐ a. Точка доступу, маска, IP-config
- ☐ b. Комутатор, TCP, маска
- ☒ c. Мережева карта, комутатор, маршрутизатор
- ☐ d. Маршрутизатор, HTTP, Uniform Resource Locator

### Question 2

Complete

Not graded

🚩 Flag question

У якому форматі користувач вводить ip адресу та маску?

Select one:

- ☐ a. 192.168.1.15
- ☐ b. 192.168.1.15 маска 24
- ☐ c. ip адреса та маска
- ☐ d. ip: 192.168.1.15 mask: 24
- ☒ e. 192.168.1.15/24

### Question 3

Complete

Not graded

🚩 Flag question

Чи можна обійтись без DNS сервера для того, щоб завантажити певну інтернет сторінку?

Select one:

- ☐ a. Можна без DNS сервера, зараз його майже ніхто не використовує

- ☐ b. Ні, не можна
- ☒ c. Можна обійтись, але з ним буде швидше

#### Question 4

Complete

Not graded

🚩 Flag question

Як працює стек протоколів?

Select one:

- ☐ a. Протоколи діляться на різні рівні. Одночасно виконуються лише перші 1000, решта у порядку черги
- ☒ b. Протоколи діляться на різні рівні, проте виконуються поступово один за одним
- ☐ c. Протоколи діляться на різні рівні, які виконуються одночасно

#### Question 5

Complete

Not graded

🚩 Flag question

Яку назву має мати функція **ЛР №2. Завдання на програмування №1**. яка повертає IP адресу?

Answer: `get_ip_from_raw_address(raw_address)`

#### Question 6

Complete

Not graded

🚩 Flag question

IP адреса конкретного пристрою не є сталою? Тобто, вона може змінюватись залежно від того, до якого інтернет-з'єднання ви підключені, коли ви підключились ітд.

Select one:

- ☒ True
- ☐ False



[Skip to main content](#)

Side panel

Enter your search query

- [Допомога викладачам](#)
- [Інструкції](#)
- [Відео-уроки](#)
- [Вебінари](#)
- [Шаблони курсів](#)
- 
- [Допомога студентам](#)
- [Інструкції з користування CMS](#)
- [Розклад занять на порталі](#)
- [Зовнішні ресурси](#)
- [Документація Moodle](#)
- [Спільнота Moodle](#)
- [YouTube канал Learn Moodle Basics 3.8](#)

0

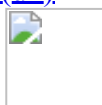
## Notifications

You have no notifications

[See all](#)

0

- - [English \(en\)](#)
  - [Українська \(uk\)](#)



- [Нагорнюк Марта](#)

[Dashboard](#)[View profile](#)[Grades](#)[Preferences](#)[Notifications](#)[Calendar](#)[Log out](#)

[Home](#)

[Dashboard](#)

[Events](#)

[My Courses](#)

[Історія європейської цивілізації \(2.2...](#)  
[Математичний аналіз 2 \(2.20-21.ПKN20/Б\).](#)  
[Основи програмування \(2.20-21.ПKN20/Б\).](#)  
[Економічний аналіз 1 \(2.20-21.ПСА20/Б\).](#)  
[Математичний аналіз \(1.20-21.ПKN20/Б-...](#)  
[Дискретна математика \(1.20-21.ПKN20/Б...](#)  
[Основи програмування \(1.20-21.ПKN20/Б...](#)  
[Історія європейської цивілізації 1 \(1...](#)  
[СЯ. 20-21.2. Світ і я. Україна: істор...](#)  
[СЯ.20-21.2. Світ і я. Філософська ети...](#)  
[СЯ.1.20-21. Вступ до університетських...](#)  
[English for 1st year students \(IT BA\)...](#)

[This course](#)

[Sections](#)

[Основи програмування](#)

[Тема 0. Вступ](#)

[Тема 1. ІНТЕРНЕТ. IP, TCP, HTTP, HTML, CSS](#)

[Тема 2. MVC, FLASK](#)

[Тема 3. Об'єктно-орієнтоване програмування. Основні поняття](#)

[Тема 4. Інкапсуляція. Успадкування. Множинне успадкування.](#)

[Тема 5. Поліморфізм. Абстрактні класи. Методи класу та статичні методи](#)  
[Тема 6. Винятки, обробка винятків, ієрархія винятків, створення власних винятків. Тестування та зневадження програм.](#)  
[Тема 7. Розроблення графічного інтерфейсу користувача та візуалізація в Python.](#)  
[Регулярні вирази.](#)  
[Попередній іспит](#)  
[Тема 8. Структури даних. Масиви.](#)  
[Тема 9. Структури даних. Зв'язані структури.](#)  
[Тема 10. Структури даних. Стек. Черга.](#)  
[Тема 11. Структури даних. Дерева. Бінарне дерево пошуку.](#)  
[Тема 12. Структури даних. Графи](#)  
[Іспит](#)  
[Групове домашнє завдання](#)

[Participants](#)

[Grades](#)

[Assignments](#)

[Quizzes](#)

[Resources](#)

[Virtual programming labs](#)

[Site administration](#)



- [Home](#)
- [Dashboard](#)
- [Events](#)
- [My Courses](#)
  - [Історія європейської цивілізації \(2.2...](#)
  - [Математичний аналіз 2 \(2.20-21.ПКН20/Б\).](#)
  - [Основи програмування \(2.20-21.ПКН20/Б\).](#)
  - [Економічний аналіз 1 \(2.20-21.ПСА20/Б\).](#)
  - [Математичний аналіз \(1.20-21.ПКН20/Б-...](#)
  - [Дискретна математика \(1.20-21.ПКН20/Б...](#)
  - [Основи програмування \(1.20-21.ПКН20/Б...](#)
  - [Історія європейської цивілізації 1 \(1...](#)
  - [СЯ. 20-21.2. Світ і я. Україна: істор...](#)
  - [СЯ.20-21.2. Світ і я. Філософська ети...](#)
  - [СЯ.1.20-21. Вступ до університетських...](#)
  - [English for 1st year students \(IT BA\)...](#)
- [This course](#)
  - [Sections](#)
    - [Основи програмування](#)
    - [Тема 0. Вступ](#)
    - [Тема 1. ІНТЕРНЕТ. IP, TCP, HTTP, HTML, CSS](#)
    - [Тема 2. MVC, FLASK](#)
    - [Тема 3. Об'єктно-орієнтоване програмування. Основні поняття](#)
    - [Тема 4. Інкапсуляція. Успадкування. Множинне успадкування.](#)
    - [Тема 5. Поліморфізм. Абстрактні класи. Методи класу та статичні методи](#)
    - [Тема 6. Винятки, обробка винятків, ієрархія винятків, створення власних винятків. Тестування та зневадження програм.](#)
    - [Тема 7. Розроблення графічного інтерфейсу користувача та візуалізація в Python.](#)  
[Регулярні вирази.](#)
    - [Попередній іспит](#)
    - [Тема 8. Структури даних. Масиви.](#)
    - [Тема 9. Структури даних. Зв'язані структури.](#)
    - [Тема 10. Структури даних. Стек. Черга.](#)
    - [Тема 11. Структури даних. Дерева. Бінарне дерево пошуку.](#)

- [Тема 12. Структури даних. Графи](#)
- [Іспит](#)
- [Групове домашнє завдання](#)
- [Participants](#)
- [Grades](#)
- [Assignments](#)
- [Quizzes](#)
- [Resources](#)
- [Virtual programming labs](#)

- Show blocks
- 
- 

1. My courses
2. [Основи програмування \(2.20-21.ПКН20/Б\)](#)
3. [Тема 2. MVC, FLASK](#)
4. [Тест до ЛР №3](#)

**Started on** Friday, 19 February 2021, 18:06

**State** Finished

**Completed on** Friday, 19 February 2021, 18:07

**Time taken** 55 secs

**Grade** 1.00 out of 1.00 (100%)

## Question 1

Correct

Mark 0.17 out of 0.17

☐  Not flaggedFlag question

### Question text

Які засоби Python дозволяють отримати/розібрати вміст html сторінки?

Select one or more:

- ☒ a. beautifulsoup
- ☒ b. requests
- ☐ c. geopy
- ☐ d. urlrequest

### Feedback

Your answer is correct.

The correct answers are: requests, beautifulsoup

## Question 2

Correct

Mark 0.16 out of 0.16

☐  Not flaggedFlag question

### Question text

В контексті даного тесту, що таке API?

Select one:

- ☒ a. Application Programming Interface
- ☐ b. Asset Priority Index
- ☐ c. Application Programmers Initiative
- ☐ d. Aircraft Platform Interface

### Feedback

Your answer is correct.

The correct answer is: Application Programming Interface

### Question 3

Correct

Mark 0.17 out of 0.17

☐  Not flaggedFlag question

### Question text

Які функції з модуля `json` слід використовувати при читанні даних у форматі `json`?

Select one or more:

- ☐ a. reads
- ☐ b. dumps
- ☒ c. load
- ☐ d. dump
- ☐ e. read
- ☒ f. loads

### Feedback

Your answer is correct.

The correct answers are: load, loads

### Question 4

Correct

Mark 0.17 out of 0.17

☐  Not flaggedFlag question

### Question text

Вкажіть відповідність між типами даних у `json` форматі та їхніми відповідниками серед типів даних Python

real number Answer 1

object Answer 2



null      Answer 3 None ▼

array      Answer 4 list ▼

### Feedback


Your answer is correct.

The correct answer is: real number → float, object → dict, null → None, array → list

### Question 5

Correct

Mark 0.16 out of 0.16

☐  Not flaggedFlag question

### Question text

Коли потрібно надсилати запит на доступ до API?

Select one:

- ☒ a. Якомога швидше, адже заявку можуть розглядати певний час  
☐ b. В день дедлайну

### Feedback

Your answer is correct.

The correct answer is: Якомога швидше, адже заявку можуть розглядати певний час

### Question 6

Correct

Mark 0.17 out of 0.17

☐  Not flaggedFlag question

### Question text

Вкажіть порядок дій в базовій схемі роботи з API

- отримання базового URL
- формування запиту згідно API
- надсилання запиту
- обробка отриманих результатів запиту

### Feedback

Your answer is correct.

Save the state of the flags

[Finish review](#)

[Previous Activity](#)

[Короткі теоретичні відомості](#)

[Next Activity](#)

[Додаткові матеріали](#)

Jump to...

[Skip Quiz navigation](#)

## Quiz navigation

 [Picture of Нагорнюк Марта](#)Нагорнюк Марта

[Question 1 This page](#) [Question 2 This page](#) [Question 3 This page](#) [Question 4 This page](#) [Question 5 This page](#) [Question 6 This page](#)

[Show one page at a time](#)[Finish review](#)

Contacts

Messages selected: 1

[Contacts 0](#)



Settings

- [Contacts](#)
- [Requests 0](#)

No contacts

No contact requests

Contact request sent

Personal space

Save draft messages, links, notes etc. to access later.

☐ Delete for me and for everyone else





No starred conversations

Group ()

No group conversations

Private ()

No private conversations

## Contacts

### Non-contacts

Load more

## Messages

Load more

No results

Search people and messages

## Privacy

You can restrict who can message you

Accept messages from:

- ☐ My contacts only
- ☐ My contacts and anyone in my courses

## Notification preferences

### General

☐ Use enter to send

Write a message...

Search

Delete selected messages

Send contact request

You have blocked this user.

Unblock user

You are unable to message this user

[See all](#)

Український католицький університет

[www.ucu.edu.ua](http://www.ucu.edu.ua)

Вул. Іл. Свенціцького, 17, м. Львів, 79011

Тел.: (38/032) 240-99-40

Центр навчальних та інноваційних технологій

[підтримка користувачів]

[www.ceit.ucu.edu.ua](http://www.ceit.ucu.edu.ua)

Ел.пошта: [schurko@ucu.edu.ua](mailto:schurko@ucu.edu.ua)

Відділ інформаційних технологій

[підтримка системи]

[www.portal.ucu.edu.ua](http://www.portal.ucu.edu.ua)

Ел. пошта: [helpdesk@ucu.edu.ua](mailto:helpdesk@ucu.edu.ua)

Усі права застережено, (с) 2015-2021. [Український Католицький Університет](#)

[Data retention summary](#)

[Get the mobile app](#)

[Skip to main content](#)

Side panel

Enter your search query

- [Допомога викладачам](#)
- [Інструкції](#)
- [Відео-уроки](#)
- [Вебінари](#)
- [Шаблони курсів](#)
- 
- [Допомога студентам](#)
- [Інструкції з користування CMS](#)
- [Розклад занять на порталі](#)
- [Зовнішні ресурси](#)
- [Документація Moodle](#)
- [Спільнота Moodle](#)
- [YouTube канал Learn Moodle Basics 3.8](#)

0

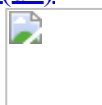
## Notifications

You have no notifications

[See all](#)

0

- - [English \(en\)](#)
  - [Українська \(uk\)](#)



- [Haronjuk Marta](#)

[Dashboard](#)[View profile](#)[Grades](#)[Preferences](#)[Notifications](#)[Calendar](#)[Log out](#)

[Home](#)

[Dashboard](#)

[Events](#)

[My Courses](#)

[Історія європейської цивілізації \(2.2...](#)  
[Математичний аналіз 2 \(2.20-21.ПКН20/Б\).](#)  
[Основи програмування \(2.20-21.ПКН20/Б\).](#)  
[Економічний аналіз 1 \(2.20-21.ПСА20/Б\).](#)  
[Математичний аналіз \(1.20-21.ПКН20/Б-...](#)  
[Дискретна математика \(1.20-21.ПКН20/Б...](#)  
[Основи програмування \(1.20-21.ПКН20/Б...](#)  
[Історія європейської цивілізації 1 \(1...](#)  
[СЯ. 20-21.2. Світ і я. Україна: істор...](#)  
[СЯ.20-21.2. Світ і я. Філософська ети...](#)  
[СЯ.1.20-21. Вступ до університетських...](#)  
[English for 1st year students \(IT BA\)...](#)

[This course](#)

[Sections](#)

[Основи програмування](#)

[Тема 0. Вступ](#)

[Тема 1. ІНТЕРНЕТ. IP, TCP, HTTP, HTML, CSS](#)

[Тема 2. MVC, FLASK](#)

[Тема 3. Об'єктно-орієнтоване програмування. Основні поняття](#)

[Тема 4. Інкапсуляція. Успадкування. Множинне успадкування.](#)

[Тема 5. Поліморфізм. Абстрактні класи. Методи класу та статичні методи](#)  
[Тема 6. Винятки, обробка винятків, ієрархія винятків, створення власних винятків. Тестування та зневадження програм.](#)  
[Тема 7. Розроблення графічного інтерфейсу користувача та візуалізація в Python.](#)  
[Регулярні вирази.](#)  
[Попередній іспит](#)  
[Тема 8. Структури даних. Масиви.](#)  
[Тема 9. Структури даних. Зв'язані структури.](#)  
[Тема 10. Структури даних. Стек. Черга.](#)  
[Тема 11. Структури даних. Дерева. Бінарне дерево пошуку.](#)  
[Тема 12. Структури даних. Графи](#)  
[Іспит](#)  
[Групове домашнє завдання](#)

[Participants](#)

[Grades](#)

[Assignments](#)

[Quizzes](#)

[Resources](#)

[Virtual programming labs](#)

[Site administration](#)



- [Home](#)
- [Dashboard](#)
- [Events](#)
- [My Courses](#)
  - [Історія європейської цивілізації \(2.2...](#)
  - [Математичний аналіз 2 \(2.20-21.ПКН20/Б\).](#)
  - [Основи програмування \(2.20-21.ПКН20/Б\).](#)
  - [Економічний аналіз 1 \(2.20-21.ПСА20/Б\).](#)
  - [Математичний аналіз \(1.20-21.ПКН20/Б-...](#)
  - [Дискретна математика \(1.20-21.ПКН20/Б...](#)
  - [Основи програмування \(1.20-21.ПКН20/Б...](#)
  - [Історія європейської цивілізації 1 \(1...](#)
  - [СЯ. 20-21.2. Світ і я. Україна: істор...](#)
  - [СЯ.20-21.2. Світ і я. Філософська ети...](#)
  - [СЯ.1.20-21. Вступ до університетських...](#)
  - [English for 1st year students \(IT BA\)...](#)
- [This course](#)
  - [Sections](#)
    - [Основи програмування](#)
    - [Тема 0. Вступ](#)
    - [Тема 1. ІНТЕРНЕТ. IP, TCP, HTTP, HTML, CSS](#)
    - [Тема 2. MVC, FLASK](#)
    - [Тема 3. Об'єктно-орієнтоване програмування. Основні поняття](#)
    - [Тема 4. Інкапсуляція. Успадкування. Множинне успадкування.](#)
    - [Тема 5. Поліморфізм. Абстрактні класи. Методи класу та статичні методи](#)
    - [Тема 6. Винятки, обробка винятків, ієрархія винятків, створення власних винятків. Тестування та зневадження програм.](#)
    - [Тема 7. Розроблення графічного інтерфейсу користувача та візуалізація в Python.](#)  
[Регулярні вирази.](#)
    - [Попередній іспит](#)
    - [Тема 8. Структури даних. Масиви.](#)
    - [Тема 9. Структури даних. Зв'язані структури.](#)
    - [Тема 10. Структури даних. Стек. Черга.](#)
    - [Тема 11. Структури даних. Дерева. Бінарне дерево пошуку.](#)



- [Тема 12. Структури даних. Графи](#)
- [Іспит](#)
- [Групове домашнє завдання](#)
- [Participants](#)
- [Grades](#)
- [Assignments](#)
- [Quizzes](#)
- [Resources](#)
- [Virtual programming labs](#)

- Show blocks
- 
- 

1. My courses
2. [Основи програмування \(2.20-21.ПКН20/Б\)](#)
3. [Тема 3. Об'єктно-орієнтоване програмування. Основні поняття](#)
4. [Тест до ЛР №4](#)

**Started on** Wednesday, 24 February 2021, 16:34

**State** Finished

**Completed on** Wednesday, 24 February 2021, 16:54

**Time taken** 19 mins 44 secs

**Grade** **0.80** out of 1.00 (**80%**)

## Question 1

Correct

Mark 0.20 out of 0.20

☐  Not flaggedFlag question

### Question text

За допомогою якого графічного примітиву ілюструють співвідношення класів як "ціле-частина", де частини не можуть існувати без цілого?

Select one:

- ☐ Узагальнення
- ☐ Агрегація
- ☐ Асоціація
- ☒ Композиція

### Feedback

Your answer is correct.

The correct answer is: Композиція

## Question 2

Incorrect

Mark 0.00 out of 0.20

☐  Not flaggedFlag question

### Question text

Кожен з об'єктів класу має спільну поведінку, однаковий набір атрибутів і спільні значення.

Select one:

- ☒ True  
☐ False

### Feedback

The correct answer is 'False'.

### Question 3

Correct

Mark 0.20 out of 0.20

☐  Not flaggedFlag question

### Question text

Який елемент графічного примітиву ілюструє таке відношення між класами, де клас «отримує» всі атрибути і методи класу, нащадком якого він є?

Select one:

- ☐ Композиція  
☐ Агрегація  
☐ Клас  
☒ Узагальнення

### Feedback

Your answer is correct.

The correct answer is: Узагальнення

### Question 4

Correct

Mark 0.20 out of 0.20

☐  Not flaggedFlag question

### Question text

Виберіть правильну відповідність до малюнків.



Answer 1 Композиція ▼



Answer 2 Агрегація ▼



Answer 3 Асоціація ▼



Answer 4 Узагальнення ▾

## Feedback

Your answer is correct.

The correct answer is:



→ Композиція,



→ Агрегація,



→ Асоціація,



→ Узагальнення

## Question 5

Correct

Mark 0.20 out of 0.20

☐ Not flaggedFlag question

## Question text

Що є унікальним екземпляром структури даних, який визначається його класом?

Select one:

- ☐ Атрибут
- ☒ Об'єкт
- ☐ Змінна
- ☐ Метод

## Feedback

Your answer is correct.

The correct answer is: Об'єкт

Save the state of the flags

[Finish review](#)

[Previous Activity](#)

[Короткі теоретичні відомості](#)

[Next Activity](#)

[ЛР №4. Завдання на програмування №1, 2](#)

Jump to... Jump to... ▾

[Skip Quiz navigation](#)

## Quiz navigation



[Picture of Нагорнюк Марта](#)Нагорнюк Марта

[Question 1 This page](#) [Question 2 This page](#) [Question 3 This page](#) [Question 4 This page](#) [Question 5 This page](#)

[Show one page at a time](#)[Finish review](#)

Contacts

Messages selected: 1 ×

Search

[Contacts 0](#)

Search



Settings

- [Contacts](#)
- [Requests 0](#)

No contacts

No contact requests

Contact request sent

Personal space

Save draft messages, links, notes etc. to access later.

☐ Delete for me and for everyone else

Block

Unblock

Remove

Add

Delete

Delete

Send contact request

Accept and add to contacts

Decline

OK

Cancel



Starred ()

No starred conversations

Group ()

No group conversations

Private ()

No private conversations

**Contacts**

**Non-contacts**

Load more

Messages

Load more

No results

Search people and messages

Privacy

You can restrict who can message you

Accept messages from:

☐ My contacts only

☐ My contacts and anyone in my courses

Notification preferences

General

☐ Use enter to send

Write a message...

Search

Delete selected messages

Send contact request

You have blocked this user.

Unblock user

You are unable to message this user

[See all](#)

Український католицький університет

[www.ucu.edu.ua](http://www.ucu.edu.ua)

Вул. Іл. Свенціцького, 17, м. Львів, 79011

Тел.: (38/032) 240-99-40

Центр навчальних та інноваційних технологій

[підтримка користувачів]

[www.ceit.ucu.edu.ua](http://www.ceit.ucu.edu.ua)

Ел.пошта: [schurko@ucu.edu.ua](mailto:schurko@ucu.edu.ua)

Відділ інформаційних технологій

[підтримка системи]

[www.portal.ucu.edu.ua](http://www.portal.ucu.edu.ua)

Ел. пошта: [helpdesk@ucu.edu.ua](mailto:helpdesk@ucu.edu.ua)

Усі права застережено, (с) 2015-2021. [Український Католицький Університет](#)

[Data retention summary](#)

[Get the mobile app](#)



[Skip to main content](#)

Side panel

Enter your search query

- [Допомога викладачам](#)
- [Інструкції](#)
- [Відео-уроки](#)
- [Вебінари](#)
- [Шаблони курсів](#)
- 
- [Допомога студентам](#)
- [Інструкції з користування CMS](#)
- [Розклад занять на порталі](#)
- [Зовнішні ресурси](#)
- [Документація Moodle](#)
- [Спільнота Moodle](#)
- [YouTube канал Learn Moodle Basics 3.8](#)

1

## Notifications

You have no notifications

[See all](#)

0

- English (en).
- Українська (uk).



- Косик Дарина

[Dashboard](#)[View profile](#)[Grades](#)[Preferences](#)[Notifications](#)[Calendar](#)[Log out](#)

[Home](#)

## Dashboard

## Events

## My Courses

[Історія європейської цивілізації \(2.2...](#)  
[Математичний аналіз 2 \(2.20-21.ПКН20/Б\).](#)  
[Дискретна математика 2 \(2.20-21.ПКН20/Б\).](#)  
[Основи програмування \(2.20-21.ПКН20/Б\).](#)  
[Математичний аналіз \(1.20-21.ПКН20/Б-...](#)  
[Дискретна математика \(1.20-21.ПКН20/Б...](#)  
[Основи програмування \(1.20-21.ПКН20/Б...](#)  
[Історія європейської цивілізації 1 \(1...](#)  
[СЯ.20-21.2. Світ і я. Біоетика](#)  
[СЯ. 20-21.2. Світ і я. Україна: істор...](#)  
[СЯ.1.20-21. Вступ до університетських...](#)  
[Computer Science English CSB \(2.20-21\).](#)  
[CS English B \(1.20-21\).](#)  
[Інформаційний ресурс для студентів І...](#)

This course

## Sections

## Основи програмування

## Тема 0. Вступ

## Тема 1. IHTEPHET. IP, TCP, HTTP, HTML, CSS

## Тема 2. MVC, FLASK

[Тема 3. Об'єктно-орієнтоване програмування. Основні поняття](#)  
[Тема 4. Інкапсуляція. Успадкування. Множинне успадкування.](#)  
[Тема 5. Поліморфізм. Абстрактні класи. Методи класу та статичні методи](#)  
[Тема 6. Винятки, обробка винятків, ієрархія винятків, створення власних винятків. Тестування та зневадження програм.](#)  
[Тема 7. Розроблення графічного інтерфейсу користувача та візуалізація в Python.](#)  
[Попередній іспит](#)  
[Тема 8. Структури даних. Масиви.](#)  
[Тема 9. Структури даних. Зв'язані структури.](#)  
[Тема 10. Структури даних. Стек. Черга.](#)  
[Тема 11. Структури даних. Дерева. Бінарне дерево пошуку.](#)  
[Тема 12. Структури даних. Графи](#)  
[Іспит](#)  
[Групове домашнє завдання](#)

[Participants](#)

[Grades](#)

[Assignments](#)

[Quizzes](#)

[Resources](#)

[Virtual programming labs](#)

[Site administration](#)



- [Home](#)
- [Dashboard](#)
- [Events](#)
- [My Courses](#)
  - [Історія європейської цивілізації \(2.2...](#)
  - [Математичний аналіз 2 \(2.20-21.ПKN20/Б\).](#)
  - [Дискретна математика 2 \(2.20-21.ПKN20/Б\).](#)
  - [Основи програмування \(2.20-21.ПKN20/Б\).](#)
  - [Математичний аналіз \(1.20-21.ПKN20/Б-...](#)
  - [Дискретна математика \(1.20-21.ПKN20/Б...](#)
  - [Основи програмування \(1.20-21.ПKN20/Б...](#)
  - [Історія європейської цивілізації 1 \(1...](#)
  - [СЯ.20-21.2. Світ і я. Біоетика](#)
  - [СЯ. 20-21.2. Світ і я. Україна: істор...](#)
  - [СЯ.1.20-21. Вступ до університетських...](#)
  - [Computer Science English CSB \(2.20-21\).](#)
  - [CS English B \(1.20-21\)](#)
  - [Інформаційний ресурс для студентів I...](#)
- [This course](#)
  - [Sections](#)
    - [Основи програмування](#)
    - [Тема 0. Вступ](#)
    - [Тема 1. ІНТЕРНЕТ. IP, TCP, HTTP, HTML, CSS](#)
    - [Тема 2. MVC, FLASK](#)
    - [Тема 3. Об'єктно-орієнтоване програмування. Основні поняття](#)
    - [Тема 4. Інкапсуляція. Успадкування. Множинне успадкування.](#)
    - [Тема 5. Поліморфізм. Абстрактні класи. Методи класу та статичні методи](#)
    - [Тема 6. Винятки, обробка винятків, ієрархія винятків, створення власних винятків. Тестування та зневадження програм.](#)
    - [Тема 7. Розроблення графічного інтерфейсу користувача та візуалізація в Python.](#)
    - [Попередній іспит](#)
    - [Тема 8. Структури даних. Масиви.](#)
    - [Тема 9. Структури даних. Зв'язані структури.](#)

- [Тема 10. Структури даних. Стек. Черга.](#)
- [Тема 11. Структури даних. Дерева. Бінарне дерево пошуку.](#)
- [Тема 12. Структури даних. Графи](#)
- [Іспит](#)
- [Групове домашнє завдання](#)
- [Participants](#)
- [Grades](#)
- [Assignments](#)
- [Quizzes](#)
- [Resources](#)
- [Virtual programming labs](#)

- Hide blocks
- 
- 

1. My courses
2. [Основи програмування \(2.20-21.ПКН20/Б\).](#)
3. [Тема 5. Поліморфізм. Абстрактні класи. Методи класу та статичні методи](#)
4. [3. Навчальний Тест №3](#)

**Started on** Thursday, 18 March 2021, 10:36

**State** Finished

**Completed on** Thursday, 18 March 2021, 10:43


**Time taken** 7 mins

**Grade** **6.00** out of 10.00 (**60%**)

## Question 1

Correct

Mark 1.00 out of 1.00

☐  Not flaggedFlag question

### Question text

Об'єкти - екземпляри абстрактних класів завжди доцільно використовувати в об'єктно-орієнтованих програмах.

Select one:

- ☐ True
- ☒ False

### Feedback

The correct answer is 'False'.

## Question 2

Correct

Mark 1.00 out of 1.00

☐  Not flaggedFlag question

### Question text

В класі нащадку абстрактного класу не обов'язково розробляти реалізації всіх абстрактних методів, які є в батьківському класі.

Select one:

- ☒ True  
☐ False

### Feedback

The correct answer is 'True'.

### Question 3

Correct

Mark 1.00 out of 1.00

☐  Not flaggedFlag question

### Question text

Чи еквівалентні ці дві реалізації property:

#1

```
@property  
def silly(self):  
    return
```

---

#2

```
def set_silly(self):  
    return
```

```
silly = property(None, set_silly)
```

Select one:

- ☐ True  
☒ False

### Feedback

The correct answer is 'False'.

### Question 4

Correct

Mark 1.00 out of 1.00

☐  Not flaggedFlag question

### Question text

Якщо для створення property скористатися декоратором @property то результат буде такий самий що при використанні функції property().

Select one:

- ☐ True  
☒ False

### Feedback

The correct answer is 'False'.

## Question 5

Correct

Mark 1.00 out of 1.00

☐  Not flaggedFlag question

### Question text

Декоратор - це функція, яка приймає функцію або метод як аргумент, додає до неї додаткову логіку і повертає змінену функцію.

Select one:

- ☒ True  
☐ False

### Feedback

The correct answer is 'True'.

## Question 6

Correct

Mark 1.00 out of 1.00

☐  Not flaggedFlag question

### Question text

Абстрактний клас містить 7 методів. Скільки з цих методів повинні мати реалізацію у класі нащадку?

Select one or more:

- ☐ 6  
☐ 2  
☐ 7  
☐ 5  
☒ 1  
☐ 0


### Feedback

Ваша відповідь правильна  
The correct answer is: 1

## Question 7

Incorrect

Mark 0.00 out of 1.00

☐  Not flaggedFlag question

### Question text

Python дозволяє доступ до атрибутів класу за допомогою оператора `.` (dot notation). Такий спосіб простий, читабельний і короткий. Також використовують методи "приховування" атрибутів, до яких обмежується доступ через крапку, але також можна створювати властивості (property) щоб до атрибутів таки можна було досягнути через крапку. Які переваги такого підходу?

Select one or more:

- ☒ Це дозволяє спростити та скоротити обсяг коду.
- ☒ Від інших програмістів буде приховано інтерфейс.
- ☐ Це дозволяє контролювати атрибути.
- ☒ Такий спосіб захищає дані та обмежує доступ до них.
- ☒ Це дозволяє використовувати декоратор `@property` та функцію `property()`

## Feedback

Ваша відповідь неправильна

The correct answers are: Такий спосіб захищає дані та обмежує доступ до них., Це дозволяє контролювати атрибути.

## Question 8

Incorrect

Mark 0.00 out of 1.00

☐  Not flaggedFlag question

## Question text

Що буде виведено на екран у результаті виконання наступного фрагмента коду:

```
class A:
    def one(self):
        return self.two()

    def two(self):
        return 'A'

class B(A):
    def two(self):
        return 'B'

obj1=A()
obj2=B()
print(obj1.two(),obj2.one())
```

Answer:


## Feedback

The correct answer is: A B

## Question 9

Not answered

Marked out of 1.00

☐  Not flaggedFlag question

### Question text

Що буде результатом виконання наступного фрагмента коду:

```
class Demo:
    def __init__(self):
        self.x = 1
    def change(self):
        self.x = 10

class Demo_derived(Demo):
    def change(self):
        self.x=self.x + 1
        return self.x

def main():
    obj = Demo_derived()
    print(obj.change())

main()
```

Answer:

### Feedback

The correct answer is: 2

## Question 10

Not answered

Marked out of 1.00

☐  Not flaggedFlag question

### Question text

Що буде результатом виконання наступного фрагмента коду:

```

class A:
    def __init__(self,x=3):
        self._x = x

class B(A):
    def __init__(self):
        super().__init__(5)

    def display(self):
        print(self._x)

def main():
    obj = B()
    obj.display()

main()

```

Answer:

## Feedback

The correct answer is: 5

Save the state of the flags

[Finish review](#)

[Previous Activity](#)

[Презентація](#)

[Next Activity](#)

[Хід виконання лабораторної роботи](#)

Jump to...

[Skip Quiz navigation](#)

## Quiz navigation

[Question 1 This page](#) [Question 2 This page](#) [Question 3 This page](#) [Question 4 This page](#) [Question 5 This page](#) [Question 6 This page](#) [Question 7 This page](#) [Question 8 This page](#) [Question 9 This page](#) [Question 10 This page](#)

[Show one page at a time](#) [Finish review](#)

Contacts

Messages selected: 1

[Contacts 0](#)



- [Contacts](#)
- [Requests 0](#)

No contacts

No contact requests

Contact request sent

Personal space

Save draft messages, links, notes etc. to access later.

☐ Delete for me and for everyone else

Block

Unblock

Remove

Add

Delete

Delete

Send contact request

Accept and add to contacts

Decline

OK

Cancel



Starred ()

No starred conversations

Group ()

No group conversations

Private ()

No private conversations

## Contacts

## Non-contacts

Load more

## Messages

Load more

No results

Search people and messages

Privacy

You can restrict who can message you

Accept messages from:

☐ My contacts only

☐ My contacts and anyone in my courses

Notification preferences

General

☐ Use enter to send

Write a message...

Search

Delete selected messages

Send contact request

You have blocked this user.

Unblock user

You are unable to message this user

[See all](#)

Український католицький університет

[www.ucu.edu.ua](http://www.ucu.edu.ua)

Вул. Іл. Свенціцького, 17, м. Львів, 79011

Тел.: (38/032) 240-99-40

Центр навчальних та інноваційних технологій

[підтримка користувачів]

[www.ceit.ucu.edu.ua](http://www.ceit.ucu.edu.ua)

Ел.пошта: [schurko@ucu.edu.ua](mailto:schurko@ucu.edu.ua)

Відділ інформаційних технологій

[підтримка системи]

[www.portal.ucu.edu.ua](http://www.portal.ucu.edu.ua)

Ел. пошта: [helpdesk@ucu.edu.ua](mailto:helpdesk@ucu.edu.ua)

Усі права застережено, (с) 2015-2021. [Український Католицький Університет](#)

[Data retention summary](#)

[Get the mobile app](#)

- Які засоби Python дозволяють отримати/розібрати вміст html сторінки? (beautifulsoup, requests)
- В контексті даного тесту, що таке API? (Application Programming Interface)
- Які функції з модуля json слід використовувати при читанні даних у форматі json? (load, loads)
- Вкажіть відповідність між типами даних у json форматі та їхніми відповідниками серед типів даних Python (real number-float null-None object-dict array-list)
- Коли потрібно надсилати запит на доступ до API? (Якомога швидше, адже заявку можуть розглядати певний час)
- Вкажіть порядок дій в базовій схемі роботи з API (отримання базового URL, формування запиту згідно API, надсилання запиту, обробка отриманих результатів запиту)
- Виберіть варіант пояснення що таке HTML, XML, JSON, який найкраще підходить для галузі веб-технологій. (Формати файлів, які використовуються для передачі даних в мережі згідно REST підходу до архітектури мережевих протоколів.)
- Що може відбутися, якщо вказати наступні параметри при запуску web-сервера, який розроблений на Flask. `app.run(debug=False, host='0.0.0.0', port=80)` (Сервер не запуститься, бо вказаний порт 80 використовує протокол HTTP, Сервер запуститься за умови, що порт 80 вільний.)
- Встановіть відповідність між призначенням та назвою програми командного рядка. ( is a console application of Windows operating systems that displays all current TCP/IP network configuration values and can modify Dynamic Host Configuration Protocol (DHCP) and Domain Name System (DNS) settings. → ipconfig, is a computer network diagnostic tool for displaying the route (path) and measuring transit delays of packets across an Internet Protocol (IP) network. → traceroute, is a computer network administration software utility used to test the reachability of a host on an Internet Protocol (IP) network. → ping, is a system administration utility in Unix-like operating systems for network interface configuration → ifconfig, is a network administration command-line tool available for many computer operating systems for querying the Domain Name System (DNS) to obtain domain name or IP address mapping or for any other specific DNS record. → nslookup)
- До яких рівнів згідно з мережевою моделлю OSI належать вказані протоколи? (HTTP → прикладний рівень, TCP → транспортний рівень, IP → мережевий рівень)
- Якщо в масці для ідентифікації вузлів є 2 байти то яка максимальна кількість вузлів може бути у мережі (підмережі). (65534)
- Об'єктно-орієнтований підхід до розроблення програмного забезпечення ґрунтується на написанні програм об'єктно-орієнтованою мовою програмування. (False)
- Який метод потрібно розробити щоб програміст отримав зручне для використання представлення екземпляра класу? (`__repr__`)
- Що буде виведено в результаті виконання такого коду `print(isinstance(str, Object))` (NameError: name 'Object' is not defined)
- Який метод першим виконується при створенні об'єкту, якщо у класі відсутній метод `__init__`? (`__init__` попереднього класу в ієрархії класів )

- За допомогою якого графічного примітиву ілюструють співвідношення класів як "ціле-частина", де частини не можуть існувати без цілого? (Композиція)
- Виберіть правильну відповідність до малюнків. (стрілочка з білим ромбом - агрегація, лінія - асоціація, стрілочка з білим трикутником - узагальнення, стрілочка з чорним ромбом - композиція)
- Який елемент графічного примітиву ілюструє таке відношення між класами, де клас «отримує» всі атрибути і методи класу, нащадком якого він є? (Узагальнення)
- Кожен з об'єктів класу має спільну поведінку, однаковий набір атрибутів і спільні значення. (False)
- Що є унікальним екземпляром структури даних, який визначається його класом? (Об'єкт)
- Яким буде результат виконання наступного фрагмента коду (class fruits obj=fruits())

```
class fruits:
    def __init__(self):
        self.price = 100
        self.__bags = 5
    def display(self):
        print(self.__bags)
obj=fruits()
obj.display()
```

(5)

- До яких атрибутів не можна доступитися безпосередньо після виконання наступного фрагмента коду? (class Demo: def \_\_init\_\_(self): \_\_a = 1)

```
class Demo:
    def __init__(self):
        __a = 1
        self.__b = 1
        self.__c__ = 1
        __d__ = 1
```

(self.\_\_b)

- Чи буде на екрані зображено 1 після виконання наступного фрагмента коду? (class

```
class Demo:
    def __init__(self):
        self.a = 1
        self.__b = 1

    def display(self):
        return self.__b

obj = Demo()
print(obj.__b)
```

Demo: obj = Demo() (False)

- Що буде виведено на екран після виконання наступного фрагмента коду? (class

```
class Demo:
    def __init__(self):
        pass

    def test(self):
        print(__name__)

obj = Demo()
obj.test()
```

Demo: obj.test() (\_\_main\_\_)

- Яка функція перевантажує оператор == ? (\_\_eq\_\_)
- Який оператор перевантажується функцією \_\_invert\_\_() ? (~)
- Який оператор перевантажується функцією \_\_or\_\_() ? (|)
- Котрі з наступних тверджень правильні? (Функція super() повертає екземпляр батьківського класу (об'єкт посередник, який дає доступ до методів попереднього класу в ієрархії класів), Виклик super().\_\_init\_\_(self, x, y) не спричинить помилку при виконанні)
- Виберіть правильні твердження про інкапсуляцію. (Інкапсуляція (encapsulation) в мові програмування Python полягає в приховуванні деталей реалізації за інтерфейсом об'єкта., В об'єкті зберігаються дані та код, який використовує ці дані а їх взаємодія прихована., Деталі реалізації об'єкту визначені в класі, який ізолює решту програми від необхідності якимось з ними стикатися.)



- Що буде виведено на екран в результаті виконання наступного фрагмента коду

```
>>> class A:
    pass
>>> class B(A):
    pass
>>> obj=B()
>>> isinstance(obj,A)
```

(class A: class B(A): isinstance(obj, A)) (True)

- Що буде виведено на екран в результаті виконання наступного фрагмента коду

```
class A:
    def __init__(self):
        self._x = 5
class B(A):
    def display(self):
        print(self._x)
def main():
    obj = B()
    obj.display()
main()
```

(class A: class B(A): main()) (5)

- Якщо клас Second є підкласом класу First то якому способу виклику методу `__init__()` класу First з класу Second потрібно віддати перевагу? (`super().__init__()`)
- Яка функція перевантажує оператор + (плюс)? (`__add__`)
- Яким буде результат виконання наступного фрагмента коду (class Demo:

```
class Demo:
    def __init__(self):
        self.a = 1
        self.__b = 1

    def get(self):
        return self.__b

obj = Demo()
print(obj.get())
```

print(obj.get())) (1)

- Яким буде результат виконання наступного фрагмента коду (class Demo:

```
class Demo:
    def __init__(self):
        self.a = 1
        self.__b = 1
    def get(self):
        return self.__b

obj = Demo()
obj.a=45
print(obj.a)
```

print(obj.a)) (45)

- Яким буде результат виконання наступного фрагмента коду (class Demo: self.a = 11

```
class Demo:
    def __init__(self):
        self.a = 11
        self.__b = 1

    def display(self):
        return self.__b

obj = Demo()
print(obj.a)
```

print(obj.a)) (11)

- Що буде виведено на екран після виконання наступного фрагмента коду? (class Fruits: obj.bags = 2 print(obj.quantity + len(obj.\_\_dict\_\_)))

```
class Fruits:
    def __init__(self, price):
        self.price = price

obj=Fruits(50)

obj.quantity=10
obj.bags=2

print(obj.quantity+len(obj.__dict__))
```

(13)

- Яка функція перевантажує оператор // ? (\_\_floordiv\_\_)
- Яка функція перевантажує оператор >> ? (\_\_rshift\_\_)

- Який клас забезпечить виведення на екран True при виконанні наступного

```
a = foo(2)
b = foo(3)
print(a < b)
```

```
class foo:
    def __init__(self, x):
        self.x = x
    def __lt__(self, other):
        if self.x < other.x:
            return True
        else:
            return False
```

фрагмента коду? (a = foo(2))

- Виберіть правильні твердження про змінну класу. Вважайте, що variable - змінна класу, а MyClass - клас. (Змінна класу є спільною для всіх екземплярів класу, Змінну класу використовують для збереження даних спільних для всіх екземплярів класу)
- Уникнути подвійного виклику методу \_\_init\_\_ (або конструктора) базового класу при diamond problem неможливо. (False)
- Чи виконається безпомилково наступний код (class Test: class Derived\_Test(Test):

```
class Test:
    def __init__(self):
        self.x = 0
class Derived_Test(Test):
    def __init__(self):
        self.y = 1
def main():
    b = Derived_Test()
    print(b.x,b.y)
main()
```

main()) (False)

- Що буде виведено на екран в результаті виконання наступного фрагмента

```
class Test:
    def __init__(self):
        self.x = 0
class Derived_Test(Test):
    def __init__(self):
        Test.__init__(self)
        self.y = 1
def main():
    b = Derived_Test()
    print(b.x,b.y)
main()
```

коду(class Test: class Derived\_Test(Test): main())

(0 1)

- Що буде виведено на екран після виконання наступного фрагмента коду (class A:

```
>>> class A:
...     def test(self):
...         print(" test of A called ")
...
... class B(A):
...     def test(self):
...         print(" test of B called ")
...
... class C(B):
...     def test1(self):
...         print(" test of C called ")
...
... class D(C):
...     def test(self):
...         print(" test of D called ")
...         super().test()
... obj=D()
... obj.test()
```

class D(C) obj = D() obj.test() (test of D called, test of B called)

- Що буде виведено на екран після виконання наступного фрагмента коду (class A:

```
class A:
    def test(self):
        print("test of A called")
class B(A):
    def test(self):
        print("test of B called")
        super().test()
class C(A):
    def test(self):
        print("test of C called")
        super().test()
class D(B,C):
    def test2(self):
        print("test of D called")
obj=D()
obj.test()
```

class D(B,C) obj.test() (test of B called, test of C called, test of A called)

- Нехай A та B це екземпляри деякого класу C. Які методи та у якій послідовності будуть викликані при виконанні print(A - B). Вкажіть через кому і без додаткових пробілів. (\_\_sub\_\_, \_\_str\_\_)

- Яким буде результат виконання наступного фрагмента коду (class student:

```
class student:
    def __init__(self):
        self.marks = 97
        self.__cgpa = 8.7
    def display(self):
        print(self.marks)
obj=student()
print(obj.__cgpa)
```

obj=student() print(obj.\_\_cgpa)) (8.7)

- Для чого можна використати успадкування від вбудованих типів? ( Щоб додати до їх інтерфейсу нові методи, Щоб створити методи з такою ж імплементацією (реалізацією), як і в базовому типі, але з іншими назвами, Створити аналог базового класу, змінивши реалізацію методів так щоб вони використовували інші алгоритми, Створити клас точно такий самий як базовий, але з іншою назвою)
- Що буде виведено на екран в результаті виконання наступного фрагмента коду

```
class A:
    def __init__(self, x= 1):
        self.x = x
class der(A):
    def __init__(self,y = 2):
        super().__init__()
        self.y = y
def main():
    obj = der()
    print(obj.x, obj.y)
main()
```

(class A: class der(A): main()) main() (1 2)

- Що буде виведено на екран в результаті виконання наступного фрагмента коду

```
class A:
    def __init__(self,x):
        self.x = x
    def count(self,x):
        self.x = self.x+1
class B(A):
    def __init__(self, y=0):
        A.__init__(self, 3)
        self.y = y
    def count(self):
        self.y += 1
def main():
    obj = B()
    obj.count()
    print(obj.x, obj.y)
main()
```

(class A: classB(A): main()) (3 1)

- Чи виконається безпомилково наступний код (class A: class B(A) main())

```
class A:
    def __init__(self):
        self.__x = 1
class B(A):
    def display(self):
        print(self.__x)
def main():
    obj = B()
    obj.display()
main()
```

(False)

- Об'єкти - екземпляри абстрактних класів завжди доцільно використовувати в об'єктно-орієнтованих програмах. (False)
- В класі нащадку абстрактного класу не обов'язково розробляти реалізації всіх абстрактних методів, які є в батьківському класі. (True)
- Якщо для створення property скористатися декоратором @property то результат буде такий самий що при використанні функції property(). (False)

- Чи еквівалентні ці дві реалізації property: (@property silly = property(set\_silly))

#1

```
@property
def silly(self):
    return
```

---

#2

```
def set_silly(self):
    return
```

```
silly = property(set_silly)
```

(True)

- Чи еквівалентні ці дві реалізації property: (@property silly = property(None, set\_silly))

#1

```
@property
def silly(self):
    return
```

---

#2

```
def set_silly(self):
    return
```

```
silly = property(None, set_silly)
```

(False)

- Абстрактний клас містить 7 методів. Скільки з цих методів повинні мати реалізацію у класі нащадку? (1)
- Python дозволяє доступ до атрибутів класу за допомогою оператора .(dot notation). Такий спосіб простий, читабельний і короткий. Також використовують методи "приховування" атрибутів, до яких обмежується доступ через крапку, але також можна створювати властивості (property) щоб до атрибутів таки можна було доступитися через крапку. Навіщо надається така можливість? (Такий спосіб захищає дані та обмежує доступ до них., Іншим програмістам буде надано зручний інтерфейс., Це дозволяє контролювати атрибути.)

- Чи успішно виконається наступний фрагмент коду: (class B: def first(): def second():)

```
class B:
    def first(self):
        print("First method called")

    def second():
        print("Second method called")

ob = B()
B.first(ob)
B.second()
```

(Так, це коректні

виклики методів)

- Що буде результатом виконання наступного фрагмента коду: (class Demo: class

```
class Demo:
    def __init__(self):
        self.x = 1

    def change(self):
        self.x = 10

class Demo_derived(Demo):
    def change(self):
        self.x = self.x + 1
        return self.x

def main():
    obj = Demo_derived()
    print(obj.change())

main()
```

Demo\_derived(Demo): main())

(2)



- Що буде результатом виконання наступного фрагмента коду: (class A: class B(A) def

```
class A:
    def __init__(self):
        self.multiply(15)
        print(self.i)

    def multiply(self, i):
        self.i = 4 * i

class B(A):
    def __init__(self):
        super().__init__()
    def multiply(self, i):
        self.i = 2 * i

obj = B()
```

multiply(self, i)) (30)

- Нехай в класі MyClass визначено метод myMethod(), метод класу myClassMethod() та статичний метод myStaticMethod(), а myExemplar - екземпляр класу MyClass й doSomething - деяка функція. Які з наведених нижче реалізацій методів містять помилки? (#Імплементация myStaticMethod)
  - @staticmethod
  - def myStaticMethod(self):
  - doSomething()
  - self.myMethod(), #Імплементация myStaticMethod
  - @staticmethod
  - def myStaticMethod():
  - doSomething()
  - myMethod(), #Імплементация myClassMethod
  - @classmethod
  - def myClassMethod():
  - doSomething()
- В класі нащадку абстрактного класу потрібно розробити реалізації всіх абстрактних методів, які є в батьківському класі. (False)

- Чи еквівалентні ці дві реалізації property: (@property silly = property(set\_silly, None,

```
#1
```

```
@property
def silly(self):
    """My silly
    """

    return
```

---

```
#2
```

```
def set_silly(self):
    return
```

```
silly = property(set_silly, None, None, "My
silly")
```

```
None, "My silly"))
(True)
```

- Декоратор - це функція, яка приймає функцію або метод як аргумент, додає до неї додаткову логіку і повертає змінену функцію. (True)

- Чи еквівалентні ці дві реалізації property: (@property silly = property(set\_silly, None,

#1

```
@property
def silly(self):
    return
```

#2

```
def set_silly(self):
    return
```

```
silly = property(set_silly, None, None, "my
silly")
```

None, "my silly"))

(False)

- Абстрактний клас містить 7 методів. Скільки з цих методів можуть бути абстрактними? ( 7, 6, 5, 1, 2)
- Python дозволяє доступ до атрибутів класу за допомогою оператора .(dot notation). Такий спосіб простий, читабельний і короткий. Також використовують методи "приховування" атрибутів, до яких обмежується доступ через крапку, але також можна створювати властивості (property) щоб до атрибутів таки можна було доступитися через крапку. Які переваги такого підходу? (Такий спосіб захищає дані та обмежує доступ до них., Це дозволяє контролювати атрибути.)
- Що буде результатом виконання наступного фрагмента коду: (class Demo: def display(self): obj = Demo() print(obj.\_\_b))

```
class Demo:
    def __init__(self):
        self.a = 1
        self.__b = 1

    def display(self):
        return self.__b

obj = Demo()
print(obj.__b)
```

(Помилка, оскільки до приватної

змінної не можна доступитися безпосередньо.)

- Нехай A та B це екземпляри класу First. Які методи будуть викликані при виконанні `print(A + B)` (`__add__()`, `__str__()`)
- Що буде результатом виконання наступного фрагмента коду: `(class A: class B(A): def main(): main()) (5)`
- Що буде виведено на екран у результаті виконання наступного фрагмента коду: `(class A: class B(A): obj1 = A() obj2 = B())`

```
class A:
    def one(self):
        return self.two()

    def two(self):
        return 'A'

class B(A):
    def two(self):
        return 'B'

obj1=A()
obj2=B()
print(obj1.two(),obj2.one())
```

(A B)

- Чи успішно виконається наступний фрагмент коду: `(ob = B() B.first(ob))`

```
class B:
    def first(self):
        print("First method called")

    def second():
        print("Second method called")

ob = B()
B.first(ob)
B.second()
```

(Так, це коректні виклики методів)

- Встановіть відповідність між регулярними виразами та рядками які їм відповідають.  
`( apythex → ^[AEIOUaeiou], apythex → [AEIOUaeiou], .apythex → [^AEIOUaeiou])`

- Встановіть відповідність між символами регулярних виразів та регулярними виразами. (`\S` → `[^\t\n\r\f\v]`, `\W` → `[^a-zA-Z0-9_]`, `\D` → `[^0-9]`, `\w` → `[a-zA-Z0-9_]`, `\s` → `[\t\n\r\f\v]`, `\d` → `[0-9]`)
- Що таке грер? (Програма командного рядка в Unix системах)
- Кожна мова програмування передбачає власний синтаксис регулярних виразів. (False)
- Чи можна за допомогою регулярних виразів знайти в тексті смайлики? (True)
- Що буде виведено на екран в результаті виконання наступного фрагмента програми? (PHP FOUND, C NOT FOUND, Python FOUND, Java FOUND, C++ FOUND)
- Яким буде результат виконання наступного фрагмента коду? (import re)

```
import re
text = 'The quickbrown fox jumps over the lazy dog.'.split()
print(' '.join([w for w in text if re.findall(r"\w{5}", w)]))
```

(quickbrown jumps)

- Елементи яких електронних адрес будуть виведені на екран в результаті виконання наступних рядків коду? (emails = pattern =)

```
import re
emails = """page26@facebook.com
page33@google.com.ua
page33@google.com
page42@gmail7.list
page42@gmail.ua
page14@true.c
"""

pattern = r'([A-Za-z0-9]+)@([A-Za-z]+)\.(\w{2,4})'
print(re.findall(pattern, emails))
```

(page26@facebook.com, page33@google.com.ua, page33@google.com, [page42@gmail.ua](mailto:page42@gmail.ua))

- Які з наступних викликів функцій модуля re дозволяють зі списку text отримати список слів з трьох літер, які починаються на літеру b (без врахування регістру)?

```
text = """Betty bought a bit of butter,
But the butter was so bitter,
So she bought some better butter,
To make the bitter butter better.""".split()
```

([w for w in text if

re.fullmatch(r'(^B|^b)\w{2,3}', w)], [w for w in text if re.fullmatch(r'(B|b)\w{,3}', w)], [w for w in text if re.findall(r'(B|b)\w{,3}', w) and len(w) == 3], [w for w in text if re.findall(r'(B|b)\w{2,3}', w) and len(w) == 3])