

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ДЕПАРТАМЕНТ ОСВІТИ І НАУКИ
ЗАПОРІЗЬКОЇ ОБЛАСНОЇ ДЕРЖАВНОЇ АДМІНІСТРАЦІЇ**

**КОМУНАЛЬНИЙ ЗАКЛАД «ЗАПОРІЗЬКИЙ ОБЛАСНИЙ ЦЕНТР
НАУКОВО-ТЕХНІЧНОЇ ТВОРЧОСТІ УЧНІВСЬКОЇ МОЛОДІ «ГРАНІ»
ЗАПОРІЗЬКОЇ ОБЛАСНОЇ РАДИ**

**Розробник
Кривенко Віка Вікторівна**

ПРОГРАМУВАННЯ МОВОЮ PYTHON

**Навчальна програма з позашкільної освіти
науково-технічного напрямку**

основний рівень – 3 роки навчання

м. Запоріжжя – 2025

Схвалено педагогічною радою комунального закладу «Запорізький обласний центр науково-технічної творчості учнівської молоді «Грані» Запорізької обласної ради, протокол № 3 від 11.09.2025 року

Затверджено наказом комунального закладу «Запорізький обласний центр науково-технічної творчості учнівської молоді «Грані» Запорізької обласної ради» від 11.09.2025 № 93

Укладач:

Кривенко Віка Вікторівна – керівник гуртка «Програмування мовою Python» відділу STEM-освіти комунального закладу «Запорізький обласний центр науково-технічної творчості учнівської молоді «Грані» Запорізької обласної ради

Рецензенти:

Андрєєв Андрій Миколайович – доцент Запорізького національного університету, завідувач кафедри загальної та прикладної фізики, професор кафедри, доктор педагогічних наук;

Смирнова Анна Василівна – керівник гуртка «Основи робототехніки та комп'ютерного моделювання» комунального закладу «Запорізький обласний центр науково-технічної творчості учнівської молоді «Грані» Запорізької обласної ради, керівник гуртка-методист

ПОЯСНЮВАЛЬНА ЗАПИСКА

У сучасному цифровому світі вміння програмувати є не лише актуальною технічною навичкою, а й потужним засобом розвитку логіки, творчості та критичного мислення. Гурток «Програмування мовою Python» створено для ознайомлення вихованців 10–18 років із принципами алгоритмізації та сучасного програмування, формування цифрових компетентностей, розвитку навичок проєктної діяльності та аналітичного мислення.

Актуальність програми зумовлена стрімким розвитком інформаційних технологій та високим попитом на фахівців, які володіють мовами програмування. Python є однією з найбільш популярних і доступних мов, що активно використовується як у наукових дослідженнях, так і в розробці прикладного та вебпрограмного забезпечення. Оволодіння Python у позашкільній освіті забезпечує основу для подальшого професійного розвитку у сфері ІТ, підтримує інтерес вихованців до технічних наук, сприяє самореалізації та професійному самовизначенню.

Програма реалізується у гуртках відділу STEM-освіти інформаційно-технічного профілю та розрахована на три роки навчання з поетапним ускладненням змісту. Заняття мають переважно практичний характер, зосереджені на створенні реальних проєктів, використанні сучасних цифрових платформ, участі у конкурсах та формуванні власного портфоліо вихованця.

Мета програми — формування у вихованців предметних і ключових компетентностей з програмування мовою Python.

Основні завдання програми полягають у формуванні таких компетентностей:

- *пізнавальної*, зміст якої полягає у засвоєнні основ алгоритмізації, синтаксису та семантики мови програмування Python, вивченні структур даних, принципів побудови програм, ознайомленні з графічними можливостями Python, основами веброзробки та наукового програмування, а також у формуванні стійкого інтересу до ІТ-сфери та комп'ютерних наук;

- *практичної*, зміст якої полягає в оволодінні навичками створення алгоритмів, написання програм на мові Python, застосування циклів, умовних операторів, функцій, модулів, об'єктно-орієнтованого підходу, роботи з файлами, базами даних, а також у використанні фреймворку Django для створення сайтів, з метою вирішення прикладних задач та реалізації власних проєктів;

- *творчої*, зміст якої полягає у розвитку здатності до самовираження через авторські програмні продукти, реалізації ідей у вигляді цифрових і вебпроектів, створенні програм з графічним інтерфейсом, застосуванні креативного підходу до побудови архітектури програм та оформлення інтерфейсів, а також у формуванні ініціативності, технічної уяви та інноваційного мислення;

- *соціальної*, зміст якої полягає у формуванні навичок командної роботи над спільними програмними проектами, презентації результатів власної діяльності, розвитку відповідальності, самостійності, критичного мислення, дотримання норм цифрової етики та правил безпечної поведінки в Інтернеті, а також у здатності ефективно спілкуватися і співпрацювати в ІТ-середовищі.

Навчальна програма «Програмування мовою Python» реалізується у гуртках науково-технічного напрямку інформаційно-технічного профілю та розрахована на 3 роки основного рівня навчання для вихованців 10-18 років:

- основний рівень, перший рік навчання, 144 год. (4 год./тиждень);
- основний рівень, другий рік навчання, 144 год. (4 год./тиждень);
- основний рівень, третій рік навчання, 144 год. (4 год./тиждень).

При проведенні теоретичних занять використовуються методи: демонстраційних прикладів, евристичної бесіди, порівняльного аналізу, широко застосовуються технічні засоби навчання, аудіовізуальні й мультимедійні технології.

На практичних заняттях використовуються метод навчальних проєктів, метод програмного навчання тощо. При вивченні засобів розробки програмного забезпечення застосовується метод відкритих програм.

Заняття поєднують теоретичний і практичний компоненти. У навчанні застосовуються індивідуальні, парні та групові форми роботи, ігрові методики, метод проєктів, участь у конкурсах, олімпіадах та освітніх хакатонах.

Оскільки програма розроблена для гуртків основного рівня навчання, вона передбачає проведення індивідуальної роботи, що відбувається відповідно до Положення про порядок організації індивідуальної та групової роботи в закладах позашкільної освіти (із змінами, внесеними згідно з Наказом МОН України № 1123 (з 1322-08) від 10.12.2008 р.). Індивідуальна робота з вихованцями проводиться у формі індивідуальних занять та консультацій. Індивідуальні заняття забезпечують особистісно- орієнтований підхід у навчанні.

Кожен навчальний рік завершується виконанням підсумкового

індивідуального або командного проєкту, що інтегрує набуті знання й навички. Вихованці створюють портфоліо з програмних рішень та проходять презентацію своїх робіт.

Програма не вимагає попередньої підготовки та побудована за принципом доступності з урахуванням вікових особливостей учнів. Можливе гнучке коригування навчального змісту відповідно до рівня підготовки вихованців без зміни цілей і завдань програми.

Програма є орієнтовною. За необхідності керівник гуртка може внести до програми зміни, які не повинні впливати на загальний зміст навчальної програми та кількість навчальних годин. Незмінними мають залишатися мета, завдання і прогнозований результат освітньої діяльності.

Основний рівень, перший рік навчання НАВЧАЛЬНО-ТЕМАТИЧНИЙ ПЛАН

<i>Тема</i>	<i>Кількість годин</i>		
	<i>теоретичних</i>	<i>практичних</i>	<i>усього</i>
Вступ	2	-	2
Розділ 1. Навчальні ресурси та інструменти програмування	4	4	8
Розділ 2. Теоретичні основи алгоритмізації	14	14	28
Розділ 3. Системи програмування. Структура програмна на мові Python. Лінійні програми	16	22	38
Розділ 4. Програмні розгалуження. Умовні оператори	8	12	20
Розділ 5. Циклічні алгоритми та програми	4	10	14
Розділ 6. Вкладені цикли. Організація складних циклічних обчислень	2	6	8
Розділ 7. Модулі в Python	2	6	8
Розділ 8. Графічні можливості Python	8	8	16
Підсумок	-	2	2

Разом	60	84	144
--------------	-----------	-----------	------------

ЗМІСТ ПРОГРАМИ

Вступ (2 год)

Теоретична частина. Мета і завдання роботи гуртка. Правила санітарії, гігієни та безпечної роботи за комп'ютером. Правила поведінки в закладі освіти, кабінеті. Інструктаж із техніки безпеки. Організаційні питання. Ознайомлення з організацією комп'ютерного класу, безпекою життєдіяльності, пожежною безпекою, основами гігієни праці під час роботи за комп'ютером.

Розділ 1. Навчальні ресурси та інструменти програмування (8 год.)

Теоретична частина. Ознайомлення з основними онлайн-ресурсами для навчання програмуванню дітей. Презентація: «Сучасні ресурси для вивчення програмування: від гри до коду». Відеоогляд освітніх платформ. Огляд середовищ для запуску програм на Python. Інтерактивна вправа: «Підбери інструмент для завдання». Онлайн-вікторина: «Вгадай середовище за скріншотом».

Практична частина. Робота з інтерактивними платформами. Розв'язання перших завдань у PylasBloques (рівні 1–6). Проходження рівнів у Blockly Games: лабіринт, пташка, черепаха. Вивчення логіки з Doodle Coding (гра на розуміння блоків). Знайомство з онлайн-редакторами коду. Створення акаунта на Replit, написання програми «Привіт, світе!». Робота в середовищі Python Fiddle – введення тексту, запуск, перегляд результату. Знайомство з інтерфейсом Online Python, базові дії. Огляд і встановлення Visual Studio Code, створення нового проєкту. Міні-завдання: Напиши перший алгоритм у блоках і текстом. Гра: «Знайди помилку в коді» – на прикладах із Replit та Blockly.

Розділ 2. Теоретичні основи алгоритмізації (28 год.)

Теоретична частина. Етапи розв'язування задач на комп'ютері. Поняття алгоритму та його властивості. Алгоритмічна система (модель обчислень), загальна схема її побудови. Види представлення алгоритмів. Базові структури алгоритмів. Приклади деяких типових найпростіших алгоритмів. Лінійний алгоритм. Алгоритм з розгалуженням. Циклічний алгоритм. Вимоги до побудови схем алгоритмів. Схема даних. Схема

програми. Схема роботи системи. Критерій ефективності алгоритмів. Алгоритмічна культура. Проєктування алгоритму та моделювання. Алгоритмізація типових обчислювальних задач.

Практична частина. Складання блок-схем найпростіших алгоритмів логічно-побутових задач.

Розділ 3. Системи програмування. Структура програмна на мові Python. Лінійні програми (38 год)

Теоретична частина. Етапи розвитку мов програмування. Загальні відомості про мови програмування. Мови низького рівня. Огляд основних мов програмування високого рівня. Сфери застосування мови Python і його місце в ієрархії засобів програмування. Установка Python на локальному комп'ютері. Онлайн засоби запуску програм на Python. Загальні відомості про Python. Переваги та недоліки мови програмування Python. Створення та запуск на виконання програм. Компіляція. IDLE. Склад системи програмування Python. Мова програмування. Порядок створення програми. Алфавіт. Основні принципи синтаксису та семантики мови Python. Типи даних. Введення та відображення даних, організація інтерфейсу програма користувач. Коментарі. Змінні в Python. Опис змінних. Оператор присвоєння. Перетворення типів. Математична та алгоритмічна модель задачі. Математичні операції. Математичні вирази. Цілочисельні та арифметичні алгоритми. Лінійні програми. Реалізація деяких фізичних моделей. Реалізація деяких фінансово-економічних моделей.

Практична частина. Реалізація простих лінійних алгоритмів та програм розв'язування математичних та логічно-побутових задач.

Розділ 4. Програмні розгалуження. Умовні оператори (20 год.)

Теоретична частина. Логічний тип. Зв'язок логічного типу з числовим типом. Логічні значення, логічні операції та вирази. Перевірка істинності. Поняття розгалуження. Схема розгалуження. Типи розгалужень, умовні та альтернативні дії. Оператори розгалуження. Організація простих розгалужень. Вкладені розгалуження.

Практична частина. Реалізація простих алгоритмів та програм з розгалуженнями. Розв'язування математичних задач з умовними та альтернативними діями.

Розділ 5. Циклічні алгоритми та програми (14 год.)

Теоретична частина. Числові діапазони, функція range(). Поняття циклу. Типи циклів. Структура циклів. Циклічні оператори. Цикли з попередньою умовою виконання. Цикли з лічильником. Переривання та продовження циклів for і while. Конфліктні ситуації при організації циклічних обчислень.

Практична частина. Реалізація циклічних програм з циклами з початковою умовою та з циклами з лічильником. Перевірка числа на простоту в Python. Задачі із знаходженням дільників числа. Розклад числа на множники в Python.

Розділ 6. Вкладені цикли. Організація складних циклічних обчислень (8 год.)

Теоретична частина. Поняття вкладеного циклу. Типи вкладених циклів. Схеми вкладених циклічних обчислень. Правила та обмеження організації вкладених циклів. Керування діями в складних циклічних обчисленнях. Метод перебору.

Практична частина. Реалізація програм з вкладеними циклами для реалізації складних математичних формул та організації складних рекурентних обчислень. Розв'язування практичних задач методом перебору.

Розділ 7. Модулі в Python (8 год.)

Теоретична частина. Модульний принцип організації програми в Python. Імпорт модулів. Способи підключення модуля зі стандартної бібліотеки. Створення свого модуля на Python. Вбудовані математичні функції. Модуль random. Функції модуля math. Робота з комплексними числами: модуль cmath. Генерація випадкових чисел.

Практична частина. Реалізація програм із базовими модулями. Створення свого модуля на Python.

Розділ 8. Графічні можливості Python (16 год.)

Теоретична частина. Поняття комп'ютерної графіки. Програмування векторної графіки. Модуль «turtle». Створення полотна. Команди руху (черепахи): вперед-назад, поворот на заданий кут, переміщення в певне положення, підняти-опустити перо, витерти. Примітиви у графічному режимі. Програмування за допомогою примітивів. Застосування for-циклу. Зміна швидкості. Використання кількох об'єктів. Зміна кольору і форми об'єкта.

Практична частина. Створення розповіді «Про мене». Створення гри

«Рок, папір, ножиці». Створення рухомої векторної графіки «Гонки черепашок». Малювання орнаментів. Створення власного графічного проєкту.

Підсумок (2 год.)

Практична частина. Презентація власних проєктів. Підбивання підсумків роботи гуртка. Аналіз вивченого матеріалу.

ПРОГНОЗОВАНИЙ РЕЗУЛЬТАТ

Вихованці мають знати:

- правила безпеки, правила санітарії та гігієни під час роботи за комп'ютером;
- призначення мови програмування та основних її елементів;
- базові поняття: алгоритм, блок-схема, змінна, константа, цикл, умова тощо;
- структуру програми на Python;
- вбудовані (базові) елементарні типи даних, арифметичні і логічні операції;
- вбудовані (базові) типи даних, послідовності, такі як, рядок (string), список (list), кортеж (tuple), множина (set), словник (dictionary) і методи роботи з ними;
- основні модулі, їх підключення і їх призначення;
- умовні вирази і розгалуження програм;
- ітератори та цикли, методику розробки циклічних програм;
- основні методи та засоби налагодження програм, порядок тестування розроблених програм;
- методику використання графічних можливостей сучасних засобів розробки програм, технологічні засоби розробки програм різного призначення;
- методику використання сучасних засобів розробки програм для розв'язування задач на обчислення.

Вихованці мають уміти:

- дотримуватись правил безпеки та безпеки життєдіяльності;
- встановлювати Python на локальному комп'ютері;
- працювати в інтерактивному режимі і в режимі сценаріїв;

- здійснювати відладку створених програм;
- розробляти програми для розв'язування прикладних задач із використанням операторів циклу, виконувати налагодження й удосконалення розроблених програм;
- використовувати різні алгоритмічні структури та змінні для розв'язання навчальних і життєвих задач;
- застосувати засоби програмування для побудови моделей.

Вихованці мають набути досвід:

- дотримання правил безпечної роботи за комп'ютером і цифрової етики;
- використання онлайн-середовищ (Replit, Python Fiddle, Blockly) для написання коду;
- побудови блок-схем та логічного планування розв'язання задач;
- створення простих програм на Python з використанням змінних, операторів, умов і циклів;
- застосування базових алгоритмів із розгалуженням, циклами та вкладеними циклами;
- підключення та використання стандартних модулів Python (math, random, turtle);
- розробки графічних мініпрограм та інтерактивних проєктів із використанням модуля Turtle;
- самостійної роботи над творчими завданнями та участі в підсумковому проєкті.

Основний рівень, другий рік навчання НАВЧАЛЬНО-ТЕМАТИЧНИЙ ПЛАН

<i>Тема</i>	<i>Кількість годин</i>		
	<i>теоретичних</i>	<i>практичних</i>	<i>усього</i>
Вступ	2	-	2
Розділ 1. Обробка текстових даних	8	18	26
Розділ 2. Алгоритми обробки одновимірних списків (масивів)	10	22	32

Розділ 3. Алгоритми обробки двохвимірних масивів	8	26	34
Розділ 4. Кортежі	4	12	16
Розділ 5. Множини	4	12	16
Розділ 6. Словники	4	12	16
Підсумок	-	2	2
Разом	40	104	144

ЗМІСТ ПРОГРАМИ

Вступ (2 год.)

Теоретична частина. Ознайомлення з програмою гуртка на рік, основними темами та очікуваними результатами. Повторення ключових понять 1-го року: змінні, типи даних, умовні оператори, цикли. Правила техніки безпеки та санітарії під час роботи за комп'ютером. Інструктаж з безпечної поведінки в комп'ютерному класі. Огляд інструментів для програмування: IDLE, Replit, Visual Studio Code, OnlinePython.

Розділ 1. Обробка текстових даних (26 год.)

Теоретична частина. Поняття текстового рядка. Опис текстових даних. Поелементна обробка текстових даних. Спеціальні символи. Апостроф. Нумерація символів. Функції обробки текстів. Операції з рядками. Логічні операції. Форматування рядків. Методи перетворення, алгоритми локалізації, декомпозиції, аналізу та перетворення текстів.

Практична частина. Реалізація алгоритмів та програм обробки текстових даних з використанням алгоритмів фрагментації, аналізу та перетворення.

Розділ 2. Алгоритми обробки одновимірних списків (масивів) (32 год)

Теоретична частина. Поняття списку (масиву). Методи опису та визначення масивів. Індеси. Модуль `numpy`. Зрізи списків. Розпакування списків. Введення та відображення масивів. Обробка елементів масиву.

Арифметичні операції та функції з масивами. Впорядкування та пошук елементів масиву за заданими властивостями. Аналіз ефективності алгоритмів впорядкування.

Практична частина. Реалізація алгоритмів та програм обробки та впорядкування одновимірних масивів.

Розділ 3. Алгоритми обробки двохвимірних масивів (34 год)

Теоретична частина. Поняття двохвимірного масиву. Способи опису та визначення. Двовимірний масив як масив із одновимірних списків. Індксація в двохвимірному масиві. Введення та відображення. Прості алгоритми обробки. Виконання векторно-матричних операцій. Приклади перетворення двохвимірних масивів. Залежна індксація при обробці двохвимірних масивів.

Практична частина. Реалізація алгоритмів та програм обробки двохвимірних масивів з використання векторно-матричних операцій для розв'язування математичних задач.

Розділ 4. Кортежі (16 год)

Теоретична частина. Поняття кортежу в Python. Основні відмінності між кортежами та списками (незмінність, структура). Синтаксис створення кортежів. Операції з кортежами: індксація, зрізи, перебір у циклі, вкладені кортежі.

Практична частина. Створення та виведення кортежів. Застосування кортежів у задачах з обчисленням середніх значень, пошуку мін/макс. Використання кортежів як ключів у словниках. Розв'язування задач на роботу з вкладеними структурами (наприклад, координати точок, дати, оцінки). Написання програм з використанням кортежів для зберігання сталих наборів даних.

Розділ 5. Множини (16 год)

Теоретична частина. Поняття множини в Python. Властивості множин: унікальність елементів, неупорядкованість. Основні операції: об'єднання, перетин, різниця, симетрична різниця. Порівняння множин із списками та кортежами.

Практична частина. Створення множин і додавання/видалення елементів. Використання множин для усунення дублікатів у списках. Розв'язування задач на знаходження спільних і різних елементів. Створення

програм з використанням множин у побутових і навчальних задачах (наприклад, аналіз списків покупок, учасників заходів).

Розділ 6. Словники (16 год)

Теоретична частина. Поняття словника. Пара «ключ – значення». Створення, додавання, зміна, видалення елементів словника. Перегляд ключів, значень, пар. Вкладені словники.

Практична частина. Написання програм зі словниками для зберігання особистих даних, оцінок, товарів тощо. Пошук, додавання, редагування даних у словниках. Складання програм з аналізом інформації за допомогою словників. Робота зі словниками як базами даних.

Підсумок (2 год.)

Практична частина. Узагальнення знань з усіх тем навчального року.

Повторення понять: списки, кортежі, множини, словники. Завершення та тестування власних програмних проєктів. Обговорення ідей на наступний навчальний рік.

ПРОГНОЗОВАНИЙ РЕЗУЛЬТАТ

Вихованці мають знати:

- поняття текстових даних, списків (масивів), кортежів, множин та словників у Python;
- базові алгоритми обробки одновимірних і двовимірних структур даних;
- основи логіки структурованого зберігання та обробки даних;
- синтаксис мови Python для реалізації алгоритмів на основі різних типів колекцій.

Вихованці мають уміти:

- реалізовувати алгоритми обробки тексту, списків, матриць, множин, словників;
- застосовувати функції, методи та бібліотеки (наприклад, numpy) для роботи з масивами;
- створювати програми з використанням вкладених структур даних;
- будувати та відлагоджувати програми середньої складності;
- презентувати власні розробки.

Вихованці мають набути досвід:

- програмування складніших задач з використанням структур даних;
- аналізу, оптимізації та тестування програмних рішень;
- самостійної роботи над індивідуальними або груповими проєктами;
- підготовки до участі у конкурсах, хакатонах або олімпіадах з Python-програмування.

**Основний рівень, третій рік навчання
НАВЧАЛЬНО-ТЕМАТИЧНИЙ ПЛАН**

<i>Тема</i>	<i>Кількість годин</i>		
	<i>теоретичних</i>	<i>практичних</i>	<i>усього</i>
Вступ	2	-	2
Розділ 1. Алгоритми на графах і комбінаторика	6	4	10
Розділ 2. Структури даних (записи, масиви записів)	2	6	8
Розділ 3. Функції в Python: звичайні та рекурсивні	10	10	20
Розділ 4. Робота з файлами та об'єктно-орієнтоване програмування	8	12	20
Розділ 5. Django: створення сайтів на Python	12	12	24
Розділ 6. Обробка даних: бази, часові ряди	8	8	16
Розділ 7. Криптографія з Python	2	6	8
Розділ 8. Олімпіадні задачі з інформатики	4	14	18
Розділ 9. Програмне забезпечення досліджень	10	6	16
Підсумок	-	2	2

Разом	64	80	144
--------------	-----------	-----------	------------

ЗМІСТ ПРОГРАМИ

Вступ (2 год.)

Теоретична частина. Ознайомлення з річним планом занять. Очікувані результати третього року навчання. Узагальнення знань попередніх років: списки, словники, функції, робота з файлами. Вступне обговорення індивідуальних чи командних проєктів, які вихованці хотіли б реалізувати.

Розділ 1. Алгоритми на графах і комбінаторика (10 год.)

Теоретична частина. Факторіал. Комбінаторні обчислення. Число перестановок. Число розміщень. Число сполук. Комбінаторика розбиттів. Алгоритми на графах. Обхід графа в ширину та глибину. Програмування алгоритмів для орієнтованих та неорієнтованих графів. Дерева.

Практична частина. Реалізація алгоритмів для комбінаторних обчислень та графів.

Розділ 2. Структури даних (записи, масиви записів) (8 год)

Теоретична частина. Поняття структурованих типів даних. Структуровані дані мови Python, записи. Створення та використання записів. Введення та відображення записів. Ієрархічні структури даних, вкладені записи. Масиви записів. Впорядкування та систематизація записів.

Практична частина. Реалізація алгоритмів та програм обробки структурованих даних. Аналіз, обробка та систематизація записів.

Розділ 3. Функції в Python: звичайні та рекурсивні (20 год)

Теоретична частина. Поняття структуризації програм. Підпрограма. Атрибути підпрограм. Форми підпрограм мови Python. Поняття функції. Особливості створення та використання функцій. Функції скалярного значення. Параметри функції. Параметри за замовчуванням. Результат функції. Контроль за виконанням функції. Особливості створення програм з використанням функцій. Оператор return. Звернення до функції. Приклади реалізації програм з використанням функцій. Функції в математичних задачах. Лямбда-функції. Вкладені функції. Параметри функціонального типу. Поняття рекурсивного алгоритму. Класифікація рекурсій. Топові задачі рекурсивних обчислень. Особливі конфліктні ситуації при використанні рекурсій. Використання параметрів. Локальні та глобальні дані в рекурсіях.

Оцінка ефективності рекурсивних алгоритмів.

Практична частина. Реалізація програм для розв'язування математичних задач з використанням функцій. Реалізація простих структурованих алгоритмів та програм без параметрів для обробки глобальних даних. Реалізація рекурсивних алгоритмів та програм для розв'язування задач комбінаторики та задач з рекурентними обчисленнями. Рекурсії в аналізі даних.

Розділ 4. Робота з файлами та об'єктно-орієнтоване програмування (20 год)

Теоретична частина. Поняття файлу даних. Класифікація файлів. Текстові та двійкові файли. Режими роботи з файлами. Запис у файли, читання файлів. Атрибути файлів. Введення з текстового файлу. Прямий доступ до записів типізованого файлу. Парадигма об'єктно-орієнтованого програмування в Python. Класи, об'єкти й екземпляри класів. Визначення класів. Конструктор і деструктор екземпляра класу. Поле об'єкта класу. Додавання й видалення полів. Методи і функції класів. Копіювання екземплярів і конструктор створення копії.

Практична частина. Реалізація алгоритмів та програм обробки текстових даних. Задачі з прямим доступом до файлових записів. Реалізація програм для створення найпростіших класів.

Розділ 5. Django: створення сайтів на Python (24 год)

Теоретична частина. Python як мова розробки web-додатків. Фреймворк Django для створення сайтів. Встановлення фреймворку. Структура Django- сайту. Створення проєкту на Django. Запуск і зупинка сервера відладки. Налаштування проєкту Django. Синхронізація з базою даних. Вбудований адміністративний сайт Django. Моделі Django. Створення моделей. Класи полів для різних типів даних. Параметри полів. Створення зв'язків. Методи та мета дані моделі. Робота з моделлю у вбудованому адміністративному веб-сайті. Видобування даних з моделей. Шаблони Django. Команди шаблонізатора. Змінні та теги шаблону. Фільтри шаблону. Рендеринг шаблону. Форми Django. Створення та використання шаблонів форм та форм, зв'язаних з моделлю. Вивантаження файлів на веб-сайт. Необхідні налаштування сайту. Зберігання файлів в моделі. Вивантаження файлів через форми.

Практична частина. Створення найпростіших сайтів з використанням Django.

Розділ 6. Обробка даних: бази, часові ряди (16 год)

Теоретична частина. Принципи роботи з базами даних засобами мови Python. Дата-фрейми, бібліотека Pandas. Обробка даних, бібліотеки GDAL, OGR, Numpy. Поняття часового ряду даних. Основні статистичні параметри часового ряду. Пробіли в даних. Поняття чистих даних. Збереження, обробка та візуалізація часових рядів даних засобами мови програмування Python.

Практична частина. Обробка даних в мові програмування Python (використання бібліотек GDAL, OGR, Numpy, Pandas). Розв'язування практичних завдань із обробкою та візуалізацією часових рядів даних засобами мови програмування Python.

Розділ 7. Криптографія з Python (8 год)

Теоретична частина. Поняття криптографії. Історія шифрування. Шифри підстановки: шифр Цезаря, афінні шифри. Принципи шифрування та дешифрування. Основи безпеки передачі інформації.

Практична частина. Створення програм для шифрування/дешифрування тексту (шифр Цезаря). Реалізація зворотного шифру, автоматичне зсувне шифрування. Вивчення ASCII-таблиці для роботи зі шрифтами та символами.

Розділ 8. Олімпіадні задачі з інформатики (18 год)

Теоретична частина. Огляд типових алгоритмів і задач, що використовуються в олімпіадах. Алгоритми сортування, пошуку, арифметичні алгоритми. Основи теорії чисел, рекурсії, теорії графів та ігор.

Практична частина. Розв'язування задач: довга арифметика, рядкові алгоритми, системи числення, діофантові рівняння, НСД/НСК, алгоритм Евкліда, задачі на послідовності, відрізки, ігрові стратегії. Самостійне розв'язання задач з тренувальних платформ (за вибором).

Розділ 9. Програмне забезпечення досліджень (16 год)

Теоретична частина. Сучасні ІТ-системи для наукових досліджень. Інформаційно-технологічна модель наукового процесу. Огляд бібліотек Python для аналітики: NumPy, Matplotlib, SciPy.

Практична частина. Створення векторів і матриць, обчислення з NumPy. Побудова графіків із Matplotlib. Розв'язання прикладних дослідницьких задач. Візуалізація статистичних даних. Робота з науковими пакетами Python для автоматизації розрахунків.

Підсумок (2 год.)

Практична частина. Підбиття підсумків навчального року. Обговорення пройденого матеріалу, оцінювання прогресу вихованців. Рефлексія та анкетування. Відзначення активних і успішних вихованців.

ПРОГНОЗОВАНИЙ РЕЗУЛЬТАТ

Вихованці мають знати:

- поняття та основи об'єктно-орієнтованого програмування (ООП) в Python;
- основні типи структур даних, включаючи записи, дерева, графи, файлові структури;
- принципи рекурсії, підпрограм, функцій, зокрема лямбда-функцій;
- основи криптографії та шифрування (шифр Цезаря, підстановки);
- основи роботи з базами даних, фреймворком Django, часовими рядами;
- можливості бібліотек Python для аналітики, візуалізації та наукових обчислень (NumPy, Pandas, Matplotlib);
- класи типових задач з олімпіад з інформатики та способи їх розв'язання.

Вихованці мають уміти:

- створювати програми з використанням функцій, рекурсій, об'єктів і класів;
- реалізовувати алгоритми для графів, дерев, комбінаторних задач;
- опрацьовувати, зберігати та виводити дані у вигляді файлів, таблиць, графіків;
- створювати прості вебзастосунки з використанням Django та моделі баз даних;
- розв'язувати прикладні задачі з використанням часових рядів, статистичних методів;
- писати програми для шифрування та дешифрування інформації;
- аналізувати та вирішувати олімпіадні задачі підвищеної складності.

Вихованці мають набути досвід:

- самостійної реалізації повноцінних програмних проєктів

(дослідницьких і прикладних);

- роботи з відкритими даними, API, візуалізації результатів;
- публічного захисту власних проєктів та презентації розроблених рішень;
- командної роботи над проєктом (використовуючи, наприклад, Git);
- підготовки до участі в олімпіадах і конкурсах з програмування.

ОРІЄНТОВНИЙ ПЕРЕЛІК ОБЛАДНАННЯ

№	Основне обладнання	Кількість шт.
1.	Персональні комп'ютери / ноутбуки	15
2.	Монітори (Full HD, за потреби)	15
3.	Мережеве обладнання (роутер/світч, точки доступу Wi-Fi)	1-2
4.	Інтерактивна дошка / мультимедійний проєктор	1
5.	Колонки / акустична система	1
6.	Принтер / БФП (багатофункціональний пристрій)	1
7.	USB-флеш-накопичувачі (для збереження/трансферу проєктів)	10

Тип програмного забезпечення	Windows
Веб-браузер	Internet Explorer, Opera
Текстовий процесор	MS Word
Векторний графічний редактор	MS PowerPoint, CorelDraw
Растровий графічний редактор	Paint, Photoshop
Середовище програмування	Python
Електронні посібники та мультимедійні курси з профільного предмета	ППЗ з реєстру МОН України
Файловий менеджер	Провідник Windows, Total Commander
Програми-архіватори	WinZip, WinRar
Табличний процесор	MS Excel
Програма для перегляду pdf документів	Adobe Acrobat Reader
Програмне забезпечення для аналізу	Matlab, Mathematica, Mathcad,

математичних даних	Maple
Редактор формул	Mathtype, MS Equation Editor, OpenOffice Math
Програми для роботи зі сканером	Програми, що йдуть в комплекті зі сканером, VueScan

ЛІТЕРАТУРА

1. Бенгтсон Б. Python. З нуля до майстра. — Київ: Наш Формат, 2022.
2. Браун М. Програмуємо на Python. Веселий вступ до програмування для дітей. — Л.: Віват, 2020.
3. Гарсія М. Python для дітей. Веселий вступ до програмування. — К.: Видавництво "Наш формат", 2021.
4. Григор'єв С. В. Мова програмування Python. Початковий курс. — К.: ВHV, 2020.
5. Крамаренко Г. В. Інформатика. Основи програмування мовою Python. — К.: Літера ЛТД, 2022.
6. Міллер Б. Основи комп'ютерних наук з Python. — К.: Видавництво «Фактор», 2021.
7. Пушкарьов В. І. Програмування Python: базовий курс. — Харків: Фоліо, 2021.
8. Рашид Е. Python для початківців. — К.: Наш формат, 2020.
9. Севостьянов О. Ю. Алгоритми та структури даних мовою Python. — Х.: Ранок, 2022.
10. Сушко М. В. Python: навчальний посібник. — К.: Академія, 2020.
11. Чан Р. Програмування на Python для підлітків. — Л.: Віват, 2022.

ІНТЕРНЕТ-РЕСУРСИ

1. Документація Python: <https://docs.python.org/3/>
2. Проєкт Code.org — <https://studio.code.org/>
3. Реальний Python — <https://realpython.com/>
4. Секрети Python — <https://python-course.eu/>