Міністерство освіти і науки України

Чернігівський національний технологічний університет

Навчально-науковий інститут електронних та інформаційних технологій

Кафедра кібербезпеки та математичного моделювання

“**ЗАТВЕРДЖУЮ**”

Завідувач кафедри

Ю.М. Ткач

“\_\_\_\_\_\_”\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_2023 р.

**РОБОЧА ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ**

**Технології програмування (ОК14)**

Освітня програма «Кібербезпека»

Рівень вищої освіти – *перший (бакалаврський)*

Спеціальність 125 - Кібербезпека та захист інформації

Мова навчання: *українська*

Статус дисципліни: *обов'язкова*

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Форма навчання | Рік навч. | Сем. | Розподіл годин | | | | | | Разом | За тиждень | | ІНДЗ | Контр. |
| Всього ауд. | Лек | Прак | Лаб. |  | СРС | Ауд. | СРС |
| Денна | 2 | 3 | 30 | 16 |  | 14 |  | 90 | 120 | 3 | 9 | РГР | Е |

Чернігів – 2023 рік

Робоча програма дисципліни *«Технології програмування»*

для здобувачів вищої освіти галузі знань *12 «Інформаційні технології»*

спеціальності *125 “Кібербезпека та захист інформації”*

Розробник робочої навчальної програми:

*викладач кафедри кібербезпеки та математичного моделювання*

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ *І.М. Дюба*

УЗГОДЖЕНО:

Гарант освітньої програми «*Кібербезпека та захист інформації*»

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ *Т.А. Петренко*

Робочу програму схвалено на засіданні кафедри *кібербезпеки та математичного моделювання*

Протокол від “*\_\_*” *\_\_\_\_\_\_\_\_\_* *20*\_\_\_ року № *\_\_*

Завідувачка кафедри *кібербезпеки та математичного моделювання*

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ *Ю.М. Ткач*

**ABSTRACT**

**Educational-Scientific Institute оf Electronic and Information Technologies**

**Programming technologies**

**2023/2024 Sem. 3**

The course aims to form an understanding of modern programming paradigms during software implementation of tasks. In the process of listening to the course, students will familiarize themselves with the principles of acquiring theoretical knowledge in the field of a new programming language, the application of theoretical knowledge in practice, conducting a surface analysis of the task and breaking it down into sub-tasks, the principles of choosing between existing options for solving the task, checking the work results.

The Python programming language was chosen for this course due to its ease of use and rapid deployment. Python is a universal language widely used around the world for a variety of purposes. Among the advantages of the Python language, it is possible to highlight the portability of written programs to computers of different architectures and with different operating systems, the brevity of writing algorithms, and the ability to obtain effective program code based on the speed of execution. Topics include basic language concepts, algorithmic structures, functional programming, handling exceptions, using libraries, writing your own code tests, working with files, and basic object-oriented programming concepts.

Among the main goals of this course is for students to gain practical experience in learning a new programming language, identify typical approaches to learning a new programming language, and use the experience to learn new programming languages in the future.

An important aspect of the course is gaining practical experience in reading someone else's code, which is the main job in industrial programming. Reading someone else's code allows you to develop abstract thinking, because during familiarization, an idea is formed about what exactly this or that software implementation does. Reading someone else's code allows you to highlight certain ideas for use in your own software implementations. When getting a job, young specialists are first of all suggested to familiarize themselves with the existing code for further modification or error correction.

# Опис навчальної дисципліни

|  |  |  |
| --- | --- | --- |
| **Найменування показників** | **Галузь знань, напрям підготовки, освітньо-кваліфікаційний рівень** | **Характеристика навчальної дисципліни** |
| **денна форма навчання** |
| Кількість кредитів – 4 | Галузь знань  *12 «Інформаційні технології»* | *Обов’язкова* |
|  |
| Модулів – *1* | Спеціальність:  *125 «Кібербезпека та захист інформації»* | **Рік підготовки:** |
| Змістових модулів – *3* | 2-й |
| Індивідуальне науково-дослідне завдання – розрахункова робота | **Семестр** |
| Загальна кількість годин – *120* | *3-й* |
| **Лекції** |
| Тижневих годин:  аудиторних – *3*;  самостійної роботи і індивідуальної студента – *9* | Освітньо-кваліфікаційний рівень:  *Бакалавр* | *16* год. |
| **Практичні, семінарські** |
| - |
| **Лабораторні** |
| 14 год. |
| **Самостійна робота** |
| 90 год. |
| **Вид контролю:** |
| Екзамен |

Примітка: Співвідношення кількості годин аудиторних занять до самостійної і індивідуальної роботи для денної форми навчання становить – 1:3.

**Застосовані скорочення:**

ЗВО – здобувач вищої освіти.

КЗ – компетентності загальні;

КФ – компетентності фахові;

ПРН – прогнозовані результати навчання;

ТП – технології програмування;

РГР – розрахунково-графічна робота.

Передумовою для вивчення дисципліни є успішне засвоєння дисциплін інформатика та бази даних.

Дисципліна є базовою для подальшого використання під час проходження різних видів практик.

# Мета навчальної дисципліни

Курс має на меті сформувати розуміння сучасних парадигм програмування під час програмної реалізації завдань. В процесі слухання курсу студенти ознайомляться з принципами здобуття теоретичних знань в галузі нової мови програмування, застосування теоретичних знання на практиці, проведення поверхневого аналізу завдання та розбивку на під задачі, принципами вибору між існуючими варіантами для вирішення завдання, перевірки результатів роботи.

Мова програмування Python вибрана для цього курсу завдяки своїй легкості використання та швидкому розгортанню. Python – це універсальна мова, що широко використовується в усьому світі для самих різних цілей. Серед переваг мови Python можна виділити переносимість написаних програм, на комп’ютери різної архітектури та з різними операційними системами, лаконічність запису алгоритмів, можливість отримати ефективний код програм за швидкістю виконання. Теми включають базові поняття мови, алгоритмічні структури, функціональне програмування, робота з винятковими ситуаціями, використання бібліотек, написання тестів власного коду, роботу з файлами та базові поняття об’єктно орієнтованого програмування.

Серед головних цілей цього курсу отримання практичного досвіду студентами в опанування нової мови програмування, виділення типових підходів для опанування нової мови програмування та використання досвіду для вивчення нових мов програмування в майбутньому.

Важливим аспектом курсу є набуття практичного досвіду в читанні чужого коду, що є основною роботу в промисловому програмуванні. Читання чужого коду дозволяє розвивати абстрактне мислення, оскільки під час ознайомлення формується уявлення про те, що саме виконує та чи інша програмна реалізація. Читання чужого коду дозволяє виділити певні ідеї для використання у власних програмних реалізаціях. При влаштування на роботу молодим спеціалістам в першу чергу пропонується ознайомитись з існуючим кодом для подальшої модифікації чи виправлення помилок.

Під час вивчення дисципліни здобувач вищої освіти має набути або розширити наступні загальні та фахові компетентності, передбачені освітньою програмою спеціальності 125 – Кібербезпека та захист інформації:

КЗ1.Здатність застосовувати знання у практичних ситуаціях.

КЗ2. Знання та розуміння предметної області та розуміння професії.

КЗ4. Вміння виявляти, ставити та вирішувати проблеми за професійним спрямуванням

# Очікувані результати навчання з дисципліни

Під час вивчення дисципліни ЗВО має досягти або вдосконалити наступні програмні результати навчання, передбачені освітньою програмою:

ПРН 15. Використовувати сучасне програмно-апаратне забезпечення інформаційно-комунікаційних технологій;

У результаті вивчення навчальної дисципліни студент повинен ***знати:***

* базові елементи та конструкції мов програмування;
* основний синтаксис мов програмування python;
* базові принципи роботи з файлами;
* принципи обробки виняткових ситуацій;
* базові принципи тестування написаного коду;
* базові принципи роботи з файлами;
* базові принципи об’єктно орієнтованого підходу;

У результаті вивчення навчальної дисципліни студент повинен ***вміти***:

* програмувати функціонально;
* представляти та обробляти інформацію;
* оцінювати правильність, структуру та стиль коду;
* вивчати нові мови програмування;
* читати документацію, робити висновки із специфікацій;
* тестувати програми , знаходити помилки та виявляти проблемні випадки;
* точно описувати симптоми помилок і чітко задавати питання.

# Критерії оцінювання результатів навчання

Для захисту виконаної лабораторної роботи ЗВО повинен продемонструвати знання та вміння застосовані при її виконанні та відповісти на питання, за вибором викладача, з лекційного курсу по темі лабораторної роботи. За кожну лабораторну роботу ЗВО отримує до 10-ти балів. При цьому враховується самостійність та своєчасність виконання лабораторної роботи, якість оформлення звіту та повнота відповідей на питання при захисті лабораторної роботи.

Для захисту виконаної РГР ЗВО повинен продемонструвати знання та вміння застосовані при її виконанні та відповісти на питання викладача. За РГР ЗВО може отримати до 35 балів. При цьому враховується самостійність та своєчасність виконання РГР, якість оформлення звіту (відповідність вимогам) та повнота відповідей на питання при захисті РГР.

З дисципліни ЗВО може набрати до 75% підсумкової оцінки за виконання всіх видів робіт, що виконуються протягом семестру. Якщо здобувач вищої освіти виконав всі види робіт, передбачених даною робочою програмою, протягом семестру (виконав, оформив та захистив всі лабораторні роботи, виконав, оформив та захистив розрахунково-графічну роботу) та отримав, в сумі, від 60 до 75 балів, то він, за бажанням, може залишити набрану кількість балів як підсумкову оцінку і не складати екзамен.

У випадку, якщо здобувач вищої освіти протягом семестру не виконав у повному обсязі передбачених цією робочою програмою всіх видів навчальної роботи, має невідпрацьовані лабораторні роботи чи РГР, має більше 30% пропусків навчальних занять без поважних причин від загального обсягу аудиторних годин відповідної навчальної дисципліни або не набрав 35 балів (кількість балів, яка сумарно з максимально можливою кількістю балів, які ЗВО може отримати під час екзамену не дозволить отримати підсумкову оцінку «задовільно – Е, 60 балів»), то він не допускається до складання екзамену під час семестрового контролю, але має право ліквідувати академічну заборгованість у порядку, передбаченому в розділі 6 [Положення про поточне та підсумкове оцінювання знань здобувачів вищої освіти Національного університету «Чернігівська політехніка».](https://stu.cn.ua/wp-content/stu-media/normobaza/normdoc/norm-osvitproces/polozhennya-pro-potochne-ta-pidsumkove-oczinyuvannya-znan-zdobuvachiv-vo.pdf)

Для складання екзамену існують білети. Білети складаються із двох питань та задачі. Якщо відповідь на питання повна і виконана задача ЗВО може отримати від 33 до 40 балів. В тому випадку, коли ЗВО виконує всі завдання без грубих помилок, він може отримати від 24 до 32 балів. Якщо при виконанні білету ЗВО допускає грубі помилки і всі питання виконані менш ніж на половину, то він може отримати від 17 до 24 балів. При невиконанні хоча б одного завдання білету, ЗВО не може отримати більше 16 балів.

Повторне складання екзамену з метою підвищення позитивної оцінки не дозволяється.

# Засоби діагностики результатів навчання

Засобами оцінювання та методами демонстрування результатів навчання:

* співбесіда із студентом по результатам лабораторних робіт, РГР;
* іспит.

# Програма навчальної дисципліни

**Тема 0. Від ідеї до імплементації**

Все починається з ідеї. Процес перетворення команд з людської мови в інструкції для обчислювальної системи. Підготовчі дії перед початком роботи: оцінка існуючих засобів, розробка архітектури, визначення набору інструментів. Вибір мови програмування, що відповідає критеріям. Основні принципи роботи інтерпретатора. Набір інструментів для роботи: інтерпретатор, середа розробки, git клієнт.

**Тема 1. Функції та змінні**

Вітаю всесвіт, як початок вивчення мові програмування. Набір базових функцій під кожну мову програмування. Аргументи для специфікації виконання функції. Можливі помилки при використанні функцій. Необмежене використання функцій. Коментарі. Результат роботи функції. Поняття змінної. Типи даних. Рядки. Числа. Булеві значення. Написання власних функцій.

**Тема 2. Умовні переходи**

Виконання інструкцій в залежності від результату відповіді на питання. Порівняння як основа запитання. Ключове слово if. Ключове слово else. Ключове слово elif. Компонування декількох питань. Ключове слово or. Ключове слово end. Змінні типу bool.

**Тема 3. Цикли**

Можливість виконувати одні і ті самі дії певну кількість раз. Ключове слово while. Ключове слово for. Складна структура даних список. Довжина списку. Основні функції для роботи зі списком. Складна структура даних словник. Комбіноване використання списків та словників. Складна структура даних кортеж. Відмінність між структурами даних.

**Тема 4. Виняткові ситуації**

Дії, що призводять до появи виняткових ситуацій. Синтаксичні помилки, що виникають в процесі написання коду. Помилки реального часу виконання. Ключове слово try. Ключове слово except. Exception як узагальнення всіх помилок. Ключове слово else для визначення інструкцій безпомилкового виконання. Ключове слово pass. Ключове слово raise.

**Тема 5. Бібліотеки**

Бібліотека як набір файлів з кодом для використовувати у розробці. Сторонні та власні бібліотеки. Модуль як механізм забезпечення повторного використання коду. Ключове слово import. Ключове слово from. Пакет як варіант сторонньої бібліотеки. Веб ресурс PyPI. Встановлення пакетів реалізовано за допомогою скрипту pip.

**Тема 6. Робота з файлами**

Механізм довготривалого зберігання та використання інформації. Функція open(). Функція close(). Функції read() та write(). Ключове слово with. Файли в форматі CSV та однойменний модуль. Основні методи роботи з CSV файлами.

**Тема 7. Об’єктно орієнтовне програмування**

Процедурний підхід написання коду. ООП – інша парадигма вирішення завдань в програмуванні. Використання абстракцій як підхід в написання коду. Можливість створення нових типів даних з вказанням імені. Ключове слово class для опису власних типів даних, що є основною характеристикою ООП. Побудова абстракція object. Атрибути класу для опису властивостей абстракції object. Можливість задавати поведінку абстракції використовуючи підходи поняття methods. Ключове слово \_\_init\_\_. Ключове слово \_\_str\_\_. Розробка власної поведінки для абстракції. Можливість посиленого контролю за атрибутами абстракції використовуючи підходи поняття properties.

# Структура навчальної дисципліни

| **Назви змістових модулів і тем** | | **Усього** | **У тому числі** | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **л** | **лаб.р.** | | | **с.р.** |
| **(денна/заочна)** | | | | |
| **Змістовий модуль №1. Основи предмету.** | | | | | | | |
| 1 | Від ідеї до імплементації | 15 | 2 |  | | 13 | |
| 2 | Функції та змінні | 15 | 2 | 2 | | 11 | |
| 3 | Умовні переходи | 15 | 2 | 2 | | 11 | |
| 4 | Цикли | 15 | 2 | 2 | | 11 | |
| 5 | Виняткові ситуації | 15 | 2 | 2 | | 11 | |
| 6 | Бібліотеки | 15 | 2 | 2 | | 11 | |
| 7 | Робота з файлами | 15 | 2 | 2 | | 11 | |
| 8 | Об’єктно орієнтовне програмування | 15 | 2 | 2 | | 11 | |
| **Разом за змістовим модулем 1** | | 120 | 16 | 2 | 11 | | |
| **Всього** | | 120 | 16 | 14 | 90 | | |

# Теми лекційних занять

|  |  |  |
| --- | --- | --- |
| **№** | **Назва теми** | **К-сть. год.**  д/з |
| 1 | Від ідеї до імплементації | 2 |
| 2 | Функції та змінні | 2 |
| 3 | Умовні переходи | 2 |
| 4 | Цикли | 2 |
| 5 | Виняткові ситуації | 2 |
| 6 | Бібліотеки | 2 |
| 7 | Робота з файлами | 2 |
| 8 | Об’єктно орієнтовне програмування | 2 |
| **Всього** | | **16** |

# Теми лабораторних робіт

|  |  |  |
| --- | --- | --- |
| **№** | **Назва теми** | **К-сть. год.** |
| 1 | Робота з списками | 2 |
| 2 | Робота з файлами | 4 |
| 3 | Об’єктно орієнтовне програмування | 4 |
| 4 | Зворотний польський запис та обчислення математичного виразу | 4 |
| **Всього** | | 14 |

# Самостійна робота

|  |  |  |
| --- | --- | --- |
| **№** | **Назва теми** | **К-сть. год.** |
| 1 | Подання інформації | 10 |
| 2 | Розгалужена структура | 10 |
| 3 | Циклічна структура | 10 |
| 4 | Рядки символів | 10 |
| 5 | Словники, кортежі | 10 |
| 6 | Рекурсія | 10 |
| 7 | Лямда функції | 10 |
| 8 | Винятки | 10 |
| 9 | ООП | 10 |
| **Всього** | | **90** |

# Індивідуальні завдання

Робочим планом передбачено виконання індивідуальних завдань у вигляді розрахунково-графічної роботи, коли ЗВО, використовуючи знання та навички отримані в процесі вивчення дисципліни на лекційних та аудиторних заняттях, самостійно виконують поставлені перед ними практичні задачі з використанням апаратного та програмного забезпечення комп’ютера.

Метою розрахунково-графічної роботи є перевірка рівня засвоєння ЗВО знань з дисципліни «Технології програмування» та вміння самостійно вирішувати поставлені перед ними практичні задачі.

Розрахунково-графічна робота виконується в другій половині третьго семестру навчання, після вивчення ЗВО найважливіших тем з дисципліни

# Методи контролю

[Оцінювання знань ЗВО здійснюється відповідно до положення «Про поточне та підсумкове оцінювання знань здобувачів вищої освіти Національного університету «Чернігівська політехніка»](https://stu.cn.ua/wp-content/stu-media/normobaza/normdoc/norm-osvitproces/polozhennya-pro-potochne-ta-pidsumkove-oczinyuvannya-znan-zdobuvachiv-vo.pdf).

Для визначення рівня засвоєння ЗВО навчального матеріалу використовуються такі методи оцінювання знань:

- поточний контроль на лабораторних заняттях (діалог, опитування, тестування, оцінка правильності самостійності та своєчасності виконання лабораторних робіт);

- оцінка за розрахунково-графічну роботу;

- підсумковий контроль (3-й семестр - екзамен).

Для діагностики знань використовується модульно-рейтингова система зі 100-бальною шкалою оцінювання.

Поточний контроль проводиться шляхом спілкування із ЗВО під час лекцій, лабораторних занять, захисту РГР, оцінювання виконання самостійної роботи.

Бали отримані під час модульного контролю дораховуються до суми балів отриманих під час поточного контролю.

Підсумковий (семестровий) контроль проводиться під час заліково-екзаменаційної сесії у формі екзамену, і оцінюється за національною шкалою та шкалою ECTS

# Розподіл балів, які отримують студенти

|  |  |  |
| --- | --- | --- |
| Форма контролю | Кількість балів | |
|
| Оцінка виконання лабораторних робіт (підготовленість, самостійність, своєчасність) | 0 | 40 |
| Розрахунково-графічна робота | 0 | 35 |
| **Підсумкова оцінка** | **0** | **75** |
| Екзамен | 0 | 25 |
| **Всього** | **0** | **100** |

**Шкала оцінювання: національна та ECTS**

|  |  |  |
| --- | --- | --- |
| Сума балів за всі види навчальної діяльності | ОцінкаECTS | Оцінка за національною шкалою |
| 90 – 100 | **А** | відмінно |
| 82-89 | **В** | добре |
| 75-81 | **С** |
| 66-74 | **D** | задовільно |
| 60-65 | **Е** |
| 0-59 | **FX** | незадовільно з можливістю повторного складання |

# Інструменти, обладнання та програмне забезпечення, використання яких передбачає навчальна дисципліна

Лекційний матеріал подається у вигляді презентацій за допомогою медіа-проектора або дистанційно в платформа MS Teams. Під час лекцій аналізуються проблемні ситуації, організовується зворотний зв'язок з аудиторією шляхом формулювання запитань і стислих відповідей з обох сторін. Особливістю виконання лабораторних робіт є застосування комп’ютерної техніки, інформаційно-комунікаційних мереж, периферійного обладнання, програмного забезпечення та технічних засобів навчальних лабораторій кафедри кібербезпеки та математичного моделювання..

# Методичне забезпечення

Методичне забезпечення навчальної дисципліни «Технології програмування» включає:

* опорний конспект лекцій на електронному носії;
* методичні вказівки з лабораторних робіт та самостійної роботи;
* основна та додаткова література;
* презентації;
* програмне забезпечення;
* питання та завдання до самостійної роботи, розрахунково-графічної роботи, модульного контролю та екзамену.

# Рекомендована література

**Базова**

1. C Programming Language. Prentice Hall, 1988. 274 p.
2. Python Basics: A Practical Introduction to Python 3 / D. Bader et al. Real Python (realpython.com), 2021. 635 p.
3. Ramalho L. Fluent Python: Clear, Concise, and Effective Programming / ed. by M. Blanchette, R. Roumeliotis. O’Reilly Media, 2015. 792 p.
4. Lutz M. Learning Python. 5th ed. 2013. 1540 p.
5. Barry P. Head First Python: A Brain-Friendly Guide. O'Reilly Media, 2016. 624 p.

# Допоміжна

1. Modern Operating Systems: Forth Edition. Pearson, 2015.
2. Matthes E. Python Crash Course, 2nd Edition: A Hands-On, Project-Based Introduction to Programming. No Starch Press, 2019. 544 p.
3. Automate the Boring Stuff with Python: Practical Programming for Total Beginners. San Francisco, USA : No Starch Press, 2015. 504 p.
4. Think Python: How to Think Like a Computer Scientist / ed. by M. Blanchette. 2nd ed. Sebastopol, California, United States of America : O’Reilly Media, 2015. 222 p.
5. Slatkin B. Effective Python: 90 Specific Ways to Write Better Python. Addison-Wesley Professional, 2019. 480 p.
6. Gorelick M., Ozsvald I. High Performance Python: Practical Performant Programming for Humans. O'Reilly Media, Incorporated, 2020. 450 p.

# Інформаційні ресурси

* Система дистанційного навчання НУ «Чернігівська політехніка». Курс: Технології програмування. – [Електронний ресурс]. – Режим доступу: <https://eln.stu.cn.ua/course/view.php?id=7199>
* Бібліотека та читальний зал НУ «Чернігівська політехніка». – [Електронний ресурс]. – Режим доступу: [http://library2.stu.cn.ua/](http://library2.stu.cn.ua/%20)
* Національна бібліотека ім В.І. Вернадського / [Електронний ресурс]. – Режим доступу: <http://www.nbuv.gov.ua/>
* Prometheus: Платформа масових відкритих онлайн-курсів [Електронний ресурс]. – Режим доступу: <https://prometheus.org.ua>