

**КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ІМЕНІ ТАРАСА ШЕВЧЕНКА**

**МЕХАНІКО-МАТЕМАТИЧНИЙ ФАКУЛЬТЕТ
кафедра інтегральних та диференціальних рівнянь**



«ЗАТВЕРДЖУЮ»
Заступник декана
з навчальної роботи
Харитонов О.М.
2020 року

РОБОЧА ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

**МОВИ ПРОГРАМУВАННЯ
для студентів**

галузь знань	11 «Математика та статистика»
спеціальність	111 «Математика»
освітній рівень	перший (бакалавр)
освітня програма	«Математика та викладання математичних дисциплін»
вид дисципліни	обов'язкова
Форма навчання	денна
	Навчальний рік 2020 /2021
	Семестр 5
	Кількість кредитів ECTS 6
	Мова викладання, навчання та оцінювання українська
	Форма заключного контролю залік

Викладачі: Перестюк Микола Олексійович, д.ф.м.н., професор, завідувач кафедри інтегральних та диференціальних рівнянь, Федоренко Юлія Володимирівна, асистент кафедри інтегральних та диференціальних рівнянь

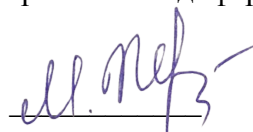
Пролонговано: на 20²¹/20²² н.р. *О. Харитонов* «31» серпня 20²¹ р.
на 20 /20 н.р. () « » 20 р.

КИЇВ – 2020

Розробники:

Перестюк Микола Олексійович, д.ф.м.н., професор, завідувач кафедри інтегральних та диференціальних рівнянь, Федоренко Юлія Володимирівна, асистент кафедри інтегральних та диференціальних рівнянь

ЗАТВЕРДЖЕНО
Зав. кафедри
інтегральних та диференціальних рівнянь



Перестюк М.О.

Протокол № 1 від 27.08.2020 р.

Схвалено науково-методичною комісією механіко-математичного факультету

Протокол № 1 від 31 серпня 2020 року

Голова науково-методичної комісії



д.ф.-м.н. Олійник А.С.

1. **Мета дисципліни** – ознайомлення з основними поняттями та положеннями з курсу «Мови програмування», здобуття навичок в програмуванні, зокрема в написанні програм на Python. Вивчення історії виникнення різних мов програмування, засоби опису даних, засоби опису дій, реалізація, параметризація, класи то об'єкти, бібліотеки програм і класів, стандарти мов програмування

2. Попередні вимоги до опанування або вибору навчальної дисципліни:

1. *Знати*: основні поняття, факти з курсів сучасне програмне забезпечення, прикладні програми, основи програмування та інформаційні технології.

2. *Вміти*: активно використовувати та творчо застосовувати зазначені вище знання в процесі опрацювання матеріалу курсу «Мови програмування», а також при написанні програм на Python, формалізувати поставлену задачу, використовувати відповідну мову програмування при розв'язанні конкретних наукових та практичних задач

3. *Володіти елементарними навичками*: написання програм, розпізнання різних мов програмування .

3. Анотація навчальної дисципліни (до 700 символів):

Навчальна дисципліна «Мови програмування» є складовою освітньої програми підготовки фахівців за освітнім рівнем «бакалавр» галузі знань 11 Математика та статистика зі спеціальності 111 Математика освітньої програми «Математика та викладання математичних дисциплін». Дана дисципліна є обов'язковою. Дисципліна «Мови програмування» вивчає виникнення мов програмування, написання програм

Викладається у 5 семестрі в обсязі 180 год. (6 кредитів ECTS¹) зокрема: лекції – 44 год., лабораторні – 44 год., консультацій – 2 год., самостійна робота – 90 год. У курсі передбачено 2 змістових модулі. Завершується дисципліна заліком.

Завдання (навчальні цілі): формування здатності розв'язувати складні спеціалізовані задачі та практичні проблеми у математиці або у процесі навчання, в галузі середньої освіти, що передбачає застосування теорій та методів математики, статистики й комп'ютерних технологій, концептуальних методів освітніх наук, психології, теорії та методики навчання і характеризується комплексністю та невизначеністю умов; набуття знань, умінь та навичок (компетентностей) на рівні новітніх досягнень у математиці, відповідно до освітнього рівня «Бакалавр». Зокрема, професійне оволодіння компетентностями:

1) Здатність застосовувати знання у практичних ситуаціях;

2) Знання й розуміння предметної області та професійної діяльності;

3) Навички використання інформаційних і комунікаційних технологій;

4) Здатність до кількісного мислення;

5) Здатність застосовувати чисельні методи для дослідження математичних моделей;

6) Здатність застосовувати спеціалізовані мови програмування та пакети прикладних програм;

7) Здатність використовувати обчислювальні інструменти для чисельних і символічних розрахунків;

8) Здатність забезпечувати розвиток прийомів розумової діяльності та просторової уяви учнів, усвідомлюючи й реалізуючи специфічні можливості процесу навчання математики для розвитку логічного та алгоритмічного мислення.

¹ кредитів ECTS – кредит кратний 30 годинам.

5. Результати навчання за дисципліною:

Результат навчання (РН) (1, знати; 2, вміти; 3, комунікація; 4, автономність та відповідальність)		Форми (та/або методи і технології) викладання і навчання	Методи оцінювання та пороговий критерій оцінювання (за необхідності)	Відсоток у підсумковій оцінці з дисципліни
Код	Результат навчання			
РН 1.1	Структура програмного забезпечення і етапи його розробки	Лекція, лабораторні роботи, самостійна робота	Активна робота на лекціях, лабораторні роботи, залік	10%
РН 1.2	Конструкція мови Python			10%
РН 1.3	Програмування на мові Python, файловий ввід-вивід			10%
РН 1.4	Структурне програмування			20%
РН 2.1	Агрегати даних	Лекція, лабораторні роботи, самостійна робота	Виконання лабораторних робіт, виконання завдань самостійної роботи, залік	10%
РН 2.2	Модульне програмування			10%
РН 2.3	Вбудовані функції			10%
РН 2.4	Розв'язувати найпростіші типи задач оптимального керування			10%
РН 3.1	Вказівники і масиви даних	лекція, самостійна робота	активна робота на лекціях	5%
РН 3.2	Вироблення навиків командної роботи			5%

6. Співвідношення результатів навчання дисципліни із програмними результатами навчання

Результати навчання дисципліни	Р	Р	Р	Р	Р	Р	Р	Р	Р	Р	Р
	Н	Н	Н	Н	Н	Н	Н	Н	Н	Н	Н
Програмні результати навчання	1	1	1	1	2	2	2	2	3	3	3

	1	2	3	4	1	2	3	4	1	2	2
<i>(з опису освітньої програми)</i>											
РН-5 - Мати навички використання спеціалізованих програмних засобів комп'ютерної та прикладної математики і використовувати інтернет-ресурси	+	+	+	+	+	+	+	+	+	+	+
РН-20 - Розв'язувати основні математичні задачі аналізу даних; застосовувати базові загальні математичні моделі для специфічних ситуацій, мати навички управління інформацією, і застосування комп'ютерних засобів статистичного аналізу даних	+	+	+	+	+	+	+	+	+	+	+

7. Схема формування оцінки.

7.1. Форми оцінювання студентів:

- оцінювання впродовж навчального періоду:

1. Активна робота на лекціях: РН1.1-РН1.4, РН3.1, РН3.2 – 5 балів/3 бали;
2. Виконання завдань для самостійної роботи: РН2.1 - РН2.4 – 15 балів/9 балів;
3. Захист лабораторних робіт 1: РН1.1, РН1.2, РН2.1, РН2.2 – 15 балів/9 балів;

4. *Захист лабораторних робіт 2*: РН1.3, РН1.4, РН2.3, РН2.4 – 15 балів/9 балів;

5. *Лабораторні заняття*: РН2.1-РН2.4, РН3.1,РН3.2–10 балів/5 балів;

Разом: 60/35

- **підсумкове оцінювання: залік.**

- *максимальна кількість балів, які можуть бути отримані*: 40 балів;

- *результати навчання, які будуть оцінюватись*: РН1.1-РН1.4, РН2.1-РН2.4

- *форма проведення і види завдань*: письмова робота.

7.2. Організація оцінювання:

Активна робота на лекціях передбачає усні відповіді на запитання лектора щодо раніше розглянутого теоретичного матеріалу.

Самостійна робота передбачає опрацювання певного обсягу теоретичного та практичного матеріалу за запропонованими джерелами.

Модульна контрольна робота проводиться письмово і складається з теоретичних завдань та типових задач за пройденим матеріалом.

Критично-розрахунковий мінімум балів за навчання впродовж семестру становить **20** балів, рекомендований мінімум, розрахований з урахуванням специфіки дисципліни становить **35** балів. Студенти, які протягом семестру набрали сумарно меншу кількість балів ніж рекомендований мінімум **35** балів для підвищення балів отримують можливість написати додаткову контрольну роботу та доскласти домашні завдання. Студенти, які набрали впродовж семестру та за рахунок додаткових етапів оцінювання сумарно меншу кількість балів ніж критично-розрахунковий мінімум – **20** балів, до складання заліку не допускаються.

Форма заліку – письмово-усна. Білет складається із 4 завдань, перші два з яких є теоретичними, два інших – задачі. Кожне завдання оцінюється від 0 до 8 балів. Додатково від 0 до 8 балів студент отримує за усне опитування. Всього за залік можна отримати від 0 до 40 балів. Мінімальна кількість балів, які додаються до семестрових – 24 бали.

Терміни проведення форм оцінювання:

1. *Захист лабораторних робіт №1*: на 6-му тижні 5 семестру.

2. *Захист лабораторних робіт №2*: на 12-му тижні 5 семестру.

3. *Оцінювання завдань самостійної роботи за РН2.1-РН2.2 на 6-му тижні, за РН2.3-РН2.4 - на 13 тижні 5 семестру.*

У випадку відсутності студента з поважних причин відпрацювання та перездачі форм контролю здійснюються у відповідності до „Положення про організацію освітнього процесу в Київському національному університеті імені Тараса Шевченка” (2018), <http://www.univ.kiev.ua/pdfs/official/Organization-of-the-educational-process.pdf>.

7.3. Шкала відповідності оцінок

Відмінно / Excellent	90-100
Добре / Good	75-89
Задовільно / Satisfactory	60-74
Незадовільно / Fail	0-59

8. СТРУКТУРА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ. ТЕМАТИЧНИЙ ПЛАН ЛЕКЦІЙ І ЛАБОРАТОРНИХ РОБІТ

теми	Назва теми	Кількість годин				
		Лекції	Лабораторні роботи	Самост. робота	Консультації	Інші форми контролю
Змістовий модуль 1 «Структура програмного забезпечення»						
1	Системне і спеціальне ПО	6	6	15		
2	Конструкція мови Python	6	6	15		
3	Програмування на мові Python, файловий ввід-вивід	10	10	15	1	
Змістовий модуль 2 «Структурне програмування»						
4	Структурування програм	6	6	15		
5	Агрегати програм	6	6	15	1	
6	Модульне програмування, вбудовані функції	10	10	15		
Всього годин		44	44	90	2	

ЗА НАВЧАЛЬНИМ ПЛАНОМ

Загальний обсяг 180 годин, у тому числі:

лекції – 44 годин,

лабораторні роботи – 44 годин,

консультації – 2 години,

самостійна робота – 90 годин.

9. Рекомендовані джерела

Основні:

1. Гавриков, М.М. Теоретические основы разработки и реализации языков программирования / М.М. Гавриков, А.Н. Иванченко. - М.: КноРус, 2018. - 207 с.
2. Александреску, А. Язык программирования D / А. Александреску. — СПб.: Символ-плюс, 2017. — 544 с.
3. Кауфман, В.Ш. Языки программирования. Концепции и принципы / В.Ш. Кауфман. — М.: ДМК, 2017. — 464 с.
4. «Architecture Patterns with Python: Enabling Test-Driven Development, Domain-Driven Design, and Event-Driven Microservices», Гарри Персиваль, Боб Грегори

Додаткові:

- 1) Гавриков, М.М. Теоретические основы разработки и реализации языков программирования: Учебное пособие / М.М. Гавриков, А.Н. Иванченко, Д.В. Гринченков. — М.: КноРус, 2016. — 184 с.
- 2) «Высокопроизводительный Python: практическое пособие для людей», Миша Горелик, Ян Освальд