



# МОДЕЛЮВАННЯ СИСТЕМ. КУРСОВА РОБОТА

## Робоча програма навчальної дисципліни (Силабус)

### Реквізити навчальної дисципліни

Рівень вищої освіти	<i>Перший (бакалаврський)</i>
Галузь знань	<i>12 Інформаційні технології</i>
Спеціальність	<i>121 Інженерія програмного забезпечення</i>
Освітня програма	<i>Інженерія програмного забезпечення інформаційних систем</i>
Статус дисципліни	<i>Нормативна</i>
Форма навчання	<i>заочна</i>
Рік підготовки, семестр	<i>4 курс, осінній семестр</i>
Обсяг дисципліни	<i>1 кредит (30 год)</i>
Семестровий контроль/ контрольні заходи	<i>Залік</i>
Розклад занять	
Мова викладання	<i>Українська</i>
Інформація про керівника курсу / викладачів	Керівники: <i>ст.викл. Дифучина Олександра Юріївна sashadif@gmail.com</i>
Розміщення курсу	Google Classroom: Моделювання систем <a href="https://classroom.google.com/c/NzExNjExNzkwODM3?cjc=vn2fbin">https://classroom.google.com/c/NzExNjExNzkwODM3?cjc=vn2fbin</a>

### Програма навчальної дисципліни

#### 1. Опис навчальної дисципліни, її мета, предмет вивчення та результати навчання

Вивчення дисципліни спрямовано на оволодіння технологіями розробки імітаційних моделей складних систем на основі універсальних об'єктно-орієнтованих мов програмування (Java, C++, C#, Python). Застосування моделей розглядається в контексті модулів інформаційних систем та систем прийняття рішень різного призначення. Дисципліна розвиває та удосконалює професійні навички програмування, набуті в попередні роки навчання, розвиває базові навички з розробки предметно-орієнтованих систем імітаційного моделювання та дослідницькі навички.

**Предмет** навчальної дисципліни – методи розробки програмного забезпечення для імітаційного моделювання складних систем, технології розробки імітаційних моделей та їх застосування в інформаційних системах.

**Метою** дисципліни є вивчення студентами принципів та способів розробки, реалізації та дослідження моделей складних систем, надбання навичок розробки алгоритмів імітації дискретно-подійних систем.

За результатами вивчення дисципліни студент повинен **набути досвід** з розробки та дослідження імітаційних алгоритмів моделей складних систем на базі універсальних мов програмування, а також з розробки модулів спеціалізованого програмного забезпечення для імітаційного моделювання дискретно-подійних систем.

**Програмні результати навчання студента.** Згідно з вимогами освітньо-професійної програми студент після засвоєння навчальної дисципліни повинен **знати**:

- методи та способи формалізації моделей складних систем,
- алгоритми імітації дискретно-подійних систем, їх верифікацію та валідацію,

- способи використання паралельних обчислень в моделюванні систем,
- методи визначення точності алгоритмів імітації та їх складності,
- методи експериментального дослідження імітаційних моделей систем,
- методи оптимізації дискретно-подійних систем,
- складові компоненти програмного забезпечення з моделювання систем,
- сучасні тенденції розвитку програмного забезпечення з моделювання систем.

Студент повинен **вміти**:

- складати формалізовані моделі систем,
- розробляти алгоритми імітації на основі подійного представлення функціонування системи,
- розробляти алгоритми імітації на основі представлення функціонування системи стохастичною мережею Петрі,
- розробляти алгоритми імітації з використанням Петрі-об'єктної технології,
- оцінювати точність та складність алгоритмів імітації,
- використовувати паралельні обчислення в алгоритмах імітації та експериментальному дослідженні моделей систем,
- розробляти графічні редактори мереж Петрі,
- виконувати експериментальне дослідження з моделями систем, у тому числі їх оптимізацію.

Згідно з вимогами освітньої програми вивчення дисципліни спрямоване на оволодіння студентом таких **компетентностей**:

ЗК2. Здатність застосовувати знання у практичних ситуаціях.

ФК19. Здатність до аналізу і оптимізації інформаційних систем з використанням математичних та імітаційних моделей і методів

**Програмні результати** вивчення дисципліни забезпечують такі програмні результати освітньої програми:

ПРН 32. Використовувати методи математичного та імітаційного моделювання при розробці та проектуванні інформаційних систем

## **2. Пререквізити та постреквізити дисципліни (місце в структурно-логічній схемі навчання за відповідною освітньою програмою)**

Вивчення дисципліни спирається на знання, отримані студентами при вивченні дисциплін «Основи програмування. Частина 2. Модульне програмування», «Теорія ймовірності», «Технології паралельних обчислень». Знання та навички, набуті студентом при вивченні дисципліни, використовуються в розробці дипломних проєктів здобувачів першого (бакалаврського) рівня вищої освіти за освітньою програмою «Інженерія програмного забезпечення інформаційних систем».

## **3. Зміст навчальної дисципліни**

Передбачено виконання курсової роботи за темою «Імітаційна модель \_\_\_\_\_ (вказуєте модель у відповідності до завдання) на основі \_\_\_\_\_ (вказуєте, який формалізм застосували при розробці)». Виконання курсової роботи передбачає розробку імітаційного алгоритму у відповідності до обраного варіанту та експериментальне дослідження імітаційної моделі. Процес, для якого розробляється алгоритм імітації, та використовуваний при розробці моделі формалізм вибираються разом з викладачем у відповідності до обраного студентом рівня складності.

Для виконання курсової роботи кожному студенту видається завдання, що містить текстовий опис об'єкта моделювання і чисельні дані про змінні та параметри об'єкта моделювання, а також про впливи зовнішнього середовища, характеристики процесу функціонування об'єкта, що необхідно оцінити в процесі моделювання.

У ході виконання курсової роботи студент повинний виконати формалізацію опису об'єкта моделювання в термінах відомих математичних схем, розробити алгоритм імітації роботи

моделі, виконати перевірку алгоритму імітації, одержати статистичні оцінки заданих характеристик моделі, побудувати і провести факторний експеримент із заданою ціллю.

За результатами виконання курсової роботи оформлюється пояснювальна записка, що містить опис розробленого алгоритмічного і програмного забезпечення моделювання системи, результатів факторного експерименту з моделлю системи, рекомендацій по використанню моделі при дослідженні і розробці реальної системи, та документацію з алгоритмів і програм моделювання.

Пояснювальна записка оформлюється згідно вимог до оформлення технічної документації. Для складання пояснювальної записки рекомендуються наступні розділи:

Анотація (не менше 650 символів)

Ключові слова (не менше 3 словосполучень)

Вступ

1. Концептуальна модель системи.
2. Формалізована модель системи.
4. Алгоритмізація моделі системи та її комп'ютерна реалізація.
5. Експериментальне дослідження моделі системи.
6. Інтерпретація результатів моделювання, формулювання висновків та пропозицій.

Висновки

Додатки (лістинги програм, лістинги результатів моделювання, результатів проведення факторного експерименту)

Варіанти завдань містять опис системи, що моделюється (виробничої, обчислювальної, телекомунікаційної, інформаційної тощо), та мету моделювання.

Приклади варіантів завдання [1]:

Приклад 1. Для забезпечення надійності АСУ ТП у ній використовується два комп'ютери. Перший комп'ютер виконує обробку даних про технологічний процес і генерування керуючих сигналів, а другий знаходиться в "гарячому резерві". Дані до АСУ надходять через  $10 \pm 2$  секунди, обробляються протягом 3 секунд, потім посилається керуючий сигнал, що підтримує заданий темп процесу. Якщо до моменту посилки наступного набору даних не отриманий керуючий сигнал, то інтенсивність виконання технологічного процесу зменшується вдвічі і дані посилаються через  $20 \pm 4$  секунд. Основний комп'ютер кожні 30 секунд посилає резервній ЕОМ сигнал про працездатність. Відсутність сигналу означає необхідність включення резервного комп'ютеру замість основного. Характеристики обох комп'ютерів однакові. Підключення резервного комп'ютеру займає 5 секунд, після цього він заміняє основний комп'ютер до відновлення, а процес повертається до нормального темпу. Відмови комп'ютера відбуваються через  $300 \pm 30$  секунд. Відновлення комп'ютера займає 100 секунд. Резервний комп'ютер абсолютно надійний.

Визначити середній час перебування технологічного процесу в загальмованому стані і середню кількість пропущених через відмови даних.

Приклад 2. У вузол комутації повідомлень, що складає з вхідного буфера, процесора, двох вихідних буферів і двох вихідних ліній, надходять повідомлення з двох напрямків. Повідомлення з одного напрямку надходять у вхідний буфер, обробляються в процесорі, буферуються у вихідному буфері першої лінії і передаються по вихідній лінії. Повідомлення з другого напрямку обробляються аналогічно, але передаються по другій вихідній лінії. Застосований метод контролю потоків потребує одночасної присутності в системі не більш трьох повідомлень на кожному напрямку. Повідомлення надходять через інтервали  $15 \pm 7$  мс. Час обробки в процесорі дорівнює 7 мс на повідомлення, час передачі по вихідній лінії дорівнює  $15 \pm 5$  мс. Якщо повідомлення надходить при наявності трьох повідомлень у напрямку, то воно одержує відмову.

Визначити завантаження пристроїв і ймовірність відмови в обслуговуванні через переповнення буфера напрямку. Визначити зміни у функції розподілу часу передачі при знятті обмежень, внесених методом контролю потоків.

Приклад 3. На обробку до інформаційної системи приймаються три класи завдань А, В і С. Виходячи з наявності оперативної пам'яті ЕОМ завдання класів А і В можуть виконуватися

одночасно. Тобто завдання класу А можуть виконуватися паралельно із завданням свого класу або із завданням класу В. Аналогічно, завдання класу В можуть виконуватися паралельно із завданням свого класу або із завданням класу А. Завдання класу С монополізують інформаційну систему. Завдання класу А поступають через інтервали часу, які розподілені за експоненціальним законом з інтенсивністю 0,2 завдання/хвилину, класу В - з інтенсивністю 0,066 завдання/хвилину, класу С - в середньому через 15 хвилин і виконуються: клас А –протягом інтервалу часу, що є випадковою величиною, яка розподілена за експоненціальним законом з інтенсивністю 0,25 завдання/хвилину, клас В - з інтенсивністю 0,166 завдання/хвилину, клас С - з інтенсивністю 0,083 завдання/хвилину.

Дисципліна обслуговування визначається комбінацією пріоритетів завдань. Можливі такі комбінації:

- А – вищий, В – середній, С – низький;
- В – вищий, А – середній, С – низький;
- С – вищий, В – середній, А – низький;
- С – вищий, А – середній, В – низький;
- С – вищий, А і В – низький;

Метою моделювання є визначення впливу різних дисциплін обслуговування при різних значеннях інтенсивності надходження на такі параметри інформаційної системи:

- середній час обслуговування завдання в системі;
- середня кількість завдань, що очікують обслуговування;
- середній час очікування в чергах.

Курсова робота має бути виконана та захищена в строк за 2 тижні до кінця семестру. Захист робіт з високою якістю виконання (більше 75 балів) відбувається з презентацією цієї роботи на одному з занять комп'ютерного практикуму (5-7 хвилин).

#### 4. Навчальні матеріали та ресурси

##### Базова література

1. Стеценко І.В. Моделювання систем: навч. посіб. / І.В. Стеценко; М-во освіти і науки України, Черк. держ. технол. ун-т. – Черкаси: видавництво „Маклаут”, 2011. – 501с.
2. Томашевський В. М. Моделювання систем. - К: Видавнича група ВНУ, 2005. - 352 с.
3. Stetsenko, I.V., Dyfuchyn, A.: Petri-object Simulation: Technique and Software. Information, Computing and Intelligent Systems 1, 51-59 (2020). ISSN 2708-4930 <https://doi.org/10.20535/2708-4930.1.2020.216057>
4. Stetsenko I.V., Dyfuchyna O. (2019) Simulation of Multithreaded Algorithms Using Petri-Object Models. Advances in Intelligent Systems and Computing, vol 754. P.391-401. [https://doi.org/10.1007/978-3-319-91008-6\\_39](https://doi.org/10.1007/978-3-319-91008-6_39)

##### Допоміжна література

1. Kelton W.D. Simulation with Arena / W.D. Kelton, R.P. Sadowski, D.A. Sadowski – New York: McGraw-Hill, 1998. - 672 p.
2. Petri nets World site TGI group at the University of Hamburg, Germany [Електронний ресурс]. – Режим доступу: <http://www.informatik.uni-hamburg.de/TGI/PetriNets/>
3. Petri Nets Tools Database Quick Overview <https://www.informatik.uni-hamburg.de/TGI/PetriNets/tools/quick.html> / accessed 11/03/2017
4. Zaitsev D.A. Clans of Petri Nets: Verification of protocols and performance evaluation of networks - LAP LAMBERT Academic Publishing, 2013 - 292 p.
5. Stetsenko I.V. State equations of stochastic timed petri nets with informational relations / I.V. Stetsenko // Cybernetics and systems analysis - Vol. 48, No 5, 2012, - P.784-797.
6. Stetsenko I.V. Petri-Object Simulation: Software Package and Complexity / I.V. Stetsenko, V. Dorosh, A. Dyfuchyn // Proceedings of the 8th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS'2015), Warsaw (Poland), 2015, pp. 381-385.



7. B. Zeigler, H. Praehofer, T. Gon Kim "Theory of Modeling and Simulation," New York: Academic Press, 2000.

8. Jensen K. Coloured Petri Nets: Modeling and Validation of Concurrent Systems / K.Jensen, L.Kristensen - Springer-Verlag Berlin Heidelberg, 2000 – 383p.

## Навчальний контент

### 5. Методика опанування навчальної дисципліни (освітнього компонента)

Матеріали для вивчення дисципліни розміщені викладачем на платформі **Google Classroom** (<https://classroom.google.com/c/NzExNjExNzkwODM3?cjc=vn2fbin>, код *vn2fbin*). Контент платформи доступний із будь-якого місця в мережі Інтернет зареєстрованому на курс студенту. Навчальний курс містить вказівки до виконання курсової роботи у вигляді презентації, перелік завдань до курсової роботи з градацією по рівням складності та весь теоретичний матеріал, необхідний для виконання завдання курсової роботи.

Виконання курсової роботи поділено на частини, які можуть здаватись студентом по мірі виконання курсової роботи: розробка концептуальної моделі, розробка формалізованої моделі, реалізація імітаційної моделі, проведення експериментального дослідження на моделі. Викладач має можливість вже на перших кроках виконання завдання виявити та вказати на помилки в розробці.

Рекомендації щодо термінів кожного етапу виконання курсової роботи наведені у таблиці.

Таблиця 1 – Графік виконання курсової роботи.

№ з/п	Назва етапу виконання роботи	Тиждень навчального семестру, відведений на виконання етапу роботи	Форма звітності
1	Отримання індивідуального завдання на курсову роботу	1-4	Затвердження теми
2	Розробка концептуальної модель системи	4-5	Оформлення відповідного розділу курсосї роботи
3	Розробка формалізованої моделі системи	6-7	Модуль програмного забезпечення
4	Алгоритмізація моделі системи та її комп'ютерна реалізація	8-10	Модуль програмного забезпечення
5	Експериментальне дослідження моделі системи	11-12	Результати тестування у вигляді тексту чи таблиці
6	Інтерпретація результатів моделювання, формулювання висновків та пропозицій	13	Результати експериментального дослідження у вигляді тексту чи таблиці
7	Оформлення пояснювальної записки	14-15	Текст пояснювальної записки, оформлений у відповідності до вимог
8	Захист курсової роботи.	16	Прилюдний захист з презентацією результатів курсової роботи

Виконане завдання курсової роботи студент здає у навчальний дистанційний курс на платформі Google Classroom по мірі виконання. Результати перевірки можна передивитись у коментарях до оцінки, а також у журналі дисципліни.

Курсова робота має бути виконана та захищена в строк за тиждень до кінця семестру. Захист роботи відбувається з презентацією цієї роботи (4-6 хвилин).

В умовах дистанційного навчання 2024-2025 н.р. усі види занять проводяться з використанням Zoom та Google Meet.

## **6. Самостійна робота студента/аспіранта**

Матеріали для самостійного вивчення дисципліни розміщені викладачем в електронному вигляді на платформі Google Classroom. Контент платформи доступний із будь-якого місця в мережі Інтернет. До самостійної роботи студента відноситься розробка програми у відповідності до завдання курсової роботи, робота з документацію програмного забезпечення, оформлення пояснювальної записки до курсової роботи, а також опрацювання додаткового теоретичного матеріалу та додаткової літератури. На самостійну роботу студент має витрати кількість годин, що дорівнює 1 кредиту.

## **Політика та контроль**

### **7. Політика навчальної дисципліни (освітнього компонента)**

Студент має вивчати дисципліну протягом семестру, дотримуючись календарного плану виконання завдання. Завдання студент має виконувати **самостійно і вчасно**. У разі виявлення порушення академічної доброчесності (плагіат коду, тексту, рисунку чи переліку джерел) студент відправляється на наступну перездачу.

Завдання вважається виконаним, якщо студент розробив програму, що реалізує імітацію системи у відповідності до завдання, описав розробку у відповідності до вказаних розділів курсової роботи, вчасно подав роботу на перевірку та захистив роботу з доповіддю про отримані результати.

Оцінювання студентів здійснюється згідно рейтингової оцінки рівня підготовки студентів з дисципліни. Поточний стан успішності студенти можуть бачити в електронному журналі. Рейтингова система оцінювання з кредитного модуля описана у наступному розділі робочої програми.

### **8. Види контролю та рейтингова система оцінювання результатів навчання (PCO)**

*Поточний контроль: перевірка дотримання графіку виконання курсової роботи та якості виконаних частин завдання*

*Семестровий контроль: залік*

*Умови допуску до семестрового контролю: 60 балів зі 100 за результатами поточного контролю.*

Рейтинг студента з кредитного модуля складається з балів, які він отримує за результатами:

- 1) розробки програмного забезпечення у відповідності до індивідуального завдання;
- 2) оформлення пояснювальної записки до курсової роботи;
- 3) прилюдного захисту з презентацією результатів курсової роботи.

Семестрова 100-бальна оцінка за виконання курсової роботи складається з балів, отриманих за розробку програмного забезпечення у відповідності до індивідуального завдання та оформлення пояснювальної записки. Набрана кількість балів не може перевищувати максимальну кількість балів завдання обраної складності.

Розподіл балів між частинами завдання:

№ з/п	Назва розділу курсової роботи	Всього	Кількість балів по видах діяльності		
			Розробка програмного забезпечення	Оформлення пояснювальної записки	Захист
1	Вступ	1	-	1	
2	Розділ 1. Концептуальна модель системи	4	-	4	-
3	Розділ 2. Формалізована модель системи	15	10	5	-
4	Розділ 3. Алгоритмізація моделі системи та її комп'ютерна реалізація	35	30	5	-
5	Розділ 4. Експериментальне дослідження моделі системи	25	20	5	-
6	Розділ 5. Інтерпретація результатів моделювання, формулювання висновків та пропозицій	5	-	5	-
7	Висновки	1	-	1	-
8	Список використаних джерел	2	-	2	-
9	Додатки	2	-	2	-
10	Захист курсової роботи	10	-	-	10
	<b>Всього балів</b>	<b>100</b>	<b>60</b>	<b>30</b>	<b>10</b>

Підсумкова оцінка переводиться до залікової оцінки згідно з таблицею:

Кількість балів	Оцінка
100-95	Відмінно
94-85	Дуже добре
84-75	Добре
74-65	Задовільно
64-60	Достатньо
Менше 60	Незадовільно
Не виконані умови допуску	Не допущено

### 9. Додаткова інформація з дисципліни (освітнього компонента)

- *Перелік спеціалізованого програмного забезпечення, що може використовуватись при виконанні курсової роботи.*

### Робочу програму навчальної дисципліни (силабус):

**Складено** професор кафедри ІПІ, д.т.н., професор Стеценко Інна Вячеславівна, ст. викладач кафедри ІПІ, доктор філософії Дифучин А.Ю., ст. викладач кафедри ІПІ, доктор філософії Дифучина О.Ю.

**Ухвалено** кафедрою ІПІ (протокол № 16 від 29.05.2024р.)

**Погоджено** Методичною комісією факультету (протокол № 10 від 21.06.2024р.)