

КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ІМЕНІ ТАРАСА ШЕВЧЕНКА

ФАКУЛЬТЕТ КОМП'ЮТЕРНИХ НАУК ТА КІБЕРНЕТИКИ

Кафедра теорії та технології програмування

«ЗАТВЕРДЖУЮ»
Заступник декана
з навчальної роботи

Олена КАШПУР
« 12 » _____ 2021 року

**РОБОЧА ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ
КОРЕКТНІСТЬ ПРОГРАМ ТА ЛОГІКИ
ПРОГРАМУВАННЯ**

для студентів

галузь знань **12 «Інформаційні технології»**
(шифр і назва)
спеціальність **122 «Комп'ютерні науки»**
(шифр і назва спеціальності)
освітній рівень **бакалавр**
(молодший бакалавр, бакалавр, магістр)
освітня програма **«Інформатика»**
(назва освітньої програми)
вид дисципліни **вибіркова**
вибірковий блок **Теорія та технологія програмування**

| | |
|--|------------|
| Форма навчання | денна |
| Навчальний рік | 2020/2021 |
| Семестр | 8 |
| Кількість кредитів ECTS | 3 |
| Мова викладання, навчання та оцінювання | українська |
| Форма заключного контролю | залік |

Викладачі: к.т.н., доцент Ткаченко О.М.

Пролонговано: на 20__/20__ н.р. _____ (_____) «__» 20__ р.
(підпис, ПІБ, дата)

на 20__/20__ н.р. _____ (_____) «__» 20__ р.
(підпис, ПІБ, дата)

Розробник: **Ткаченко Олексій Миколайович**, к. т. н., доцент кафедри теорії та технології програмування



ЗАТВЕРДЖЕНО

Зав. кафедри «Теорії та технології програмування»


_____ (підпис)

Микола НІКІТЧЕНКО
(прізвище та ініціали)

Протокол № В від «М» листопада 2021 р.

Схвалено гарантом освітньо-професійної програми «Інформатика»


_____ Людмила ОМЕЛЬЧУК «М» листопада 2021 рік
(підпис) (прізвище та ініціали)

Схвалено науково-методичною комісією факультету комп'ютерних наук та кібернетики

Протокол від «М» листопада 2021 року № 7

Голова науково-методичної комісії _____
(підпис) (прізвище та ініціали)

Людмила ОМЕЛЬЧУК

1. Мета дисципліни – засвоєння базових знань з дисципліни «Коректність програм та логіки програмування», що передбачає розгляд основних формальних методів валідації програмних систем та верифікації, а саме: тестування, формальна верифікація, перевірка моделі. Підготовка професіоналів, здатних розв'язувати завдання, що пов'язані з практичним застосуванням методів верифікації та валідації програмних систем.

2. Попередні вимоги до опанування або вибору навчальної дисципліни (за наявності):

1. *Знати:* основні поняття, відомості з програмування, теорії програмування, дискретної математики, математичної логіки та теорії алгоритмів.
2. *Вміти:* описувати синтаксис та семантику, створювати програми з використанням функціонального та логічного підходу.
3. *Володіти елементарними навичками:* програмування в сучасних мовах і середовищах.

3. Анотація навчальної дисципліни:

Навчальна дисципліна "Коректність програм та логіки програмування" є складовою освітньо-професійної програми підготовки фахівців за першим (*бакалаврським*) рівнем вищої освіти *галузі знань* 12 "Інформаційні технології" зі *спеціальності* 122 "Комп'ютерні науки", *освітньо-професійної програми* – "Інформатика".

Дана дисципліна є навчальною дисципліною за вибором за *програмою "Інформатика"* за *вибірковим блоком "Теорія та технологія програмування"*.

Викладається в 8 семестрі 4 курсу в обсязі 90 годин.

(*3 кредити ECTS*)) зокрема: *семінарські заняття – 30 год., самостійна робота – 60 год.* У курсі передбачено *2 частини та 2 контрольні роботи.* Завершується дисципліна – **заліком в 8 семестрі.**

В результаті вивчення навчальної дисципліни студент повинен:

знати: методи верифікації і валідації програмних систем, логічні засоби специфікації програм та методи перевірки на моделях;

вміти: застосовувати методи доведення правильності програм, планувати та використовувати методи тестування програмних систем.

4. Завдання (навчальні цілі):

набуття знань, умінь та навичок (компетентностей) на рівні новітніх досягнень у програмуванні, відповідно освітньої кваліфікації «Бакалавр з комп'ютерних наук».

Зокрема:

– СК1. Здатність до математичного та логічного мислення, формулювання та досліджування математичних моделей, зокрема дискретних математичних моделей, обґрунтування вибору методів і підходів для розв'язування теоретичних і прикладних задач у галузі комп'ютерних наук, аналізу та інтерпретування;

– СК17.2. Здатність формулювати та забезпечувати вимоги щодо якості програмного забезпечення у відповідності з вимогами, технічним завданням та стандартами;

– СК18.2. Здатність до алгоритмічного та логічного мислення;

– СК19.2. Здатність брати участь у проектуванні програмного забезпечення, включаючи проведення моделювання (формальний опис) його структури, поведінки та процесів функціонування.

5. Результати навчання за дисципліною:

| Результат навчання (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 | | | |
| РН1.1 | Знати необхідні методи верифікації і валідації програмних систем | Семінарські заняття | Контрольна робота 60% правильних відповідей | 20% |
| РН1.2 | Знати логічні засоби специфікації програм та методи перевірки на моделях | Семінарські заняття | Контрольна робота 60% правильних відповідей | 20% |
| РН2.1 | Вміти застосовувати методи доведення правильності програм | Семінарські заняття, самостійна робота | Захист самостійної роботи, контрольна робота | 20% |
| РН2.2 | Вміти планувати та використовувати методи тестування програмних систем | Семінарські заняття, самостійна робота | Захист самостійної роботи, контрольна робота | 20% |
| РН3.1 | Обґрунтовувати власний погляд на задачу, спілкуватися з колегами з питань проектування, розробки специфікацій та програм. | Самостійна робота | Захист самостійної роботи | 10% |
| РН4.1 | Організувати свою самостійну роботу для досягнення результату. | самостійна робота | Захист самостійної роботи | 10% |

6. Співвідношення результатів навчання дисципліни із програмними результатами навчання

| Результати навчання дисципліни | РН 1.1 | РН 1.2 | РН 2.1 | РН 2.2 | РН 3.1 | РН 4.1 |
|--|--------|--------|--------|--------|--------|--------|
| Програмні результати навчання | | | | | | |
| <i>(з опису освітньої програми)</i> | | | | | | |
| ПРН17.2. Знати та вміти застосовувати підходи щодо оцінки та забезпечення якості програмного забезпечення. | + | + | + | + | | |
| ПРН19.2. Вибирати вихідні дані для проектування, керуючись формальними методами опису вимог та моделювання. | | | + | + | + | + |

7. Схема формування оцінки.

7.1 Форми оцінювання студентів:

- семестрове оцінювання:

1. *Контрольні роботи: РН 1.1., РН 1.2, РН 2.1, РН 2.2 — 40/24 балів.*
2. *Самостійна робота: РН 2.1, РН 2.2, РН 3.1, РН 4.1 — 60/36 балів.*

- підсумкове оцінювання у формі заліку. *Виставляється за результатами роботи студентами впродовж усього семестру та не передбачає додаткових заходів оцінювання для успішних студентів.*

7.2. Організація оцінювання:

Терміни проведення форм оцінювання:

1. *Контрольна робота 1: до 5 тижня семестру.*
2. *Контрольна робота 2: до 12 тижня семестру.*

Студент має право на одне перескладання кожної контрольної роботи із можливістю отримання максимально 80% початково визначених за цю контрольну роботу балів. Термін перескладання визначається викладачем.

7.3 Шкала відповідності оцінок

| | |
|-----------------------------|---------------|
| Зараховано / Passed | 60-100 |
| Не зараховано / Fail | 0-59 |

8. Структура навчальної дисципліни. Тематичний план лекцій

| № лекції | Назва лекції | Кількість годин | |
|----------|--|-----------------|-----------|
| | | Семін. заняття | Сам. роб. |
| | Частина 1. Коректність та якість програмного забезпечення. | | |
| 1 | Тема 1. Якість програмного забезпечення. Критерії, фактори якості. Управління якістю ПЗ. <i>Самостійна робота: Знайомство зі стандартами тестування та управління якістю програмного забезпечення.</i> | 2 | 4 |
| 2 | Тема 2. Поняття та задачі валідації. Характеристика основних методів валідації систем. Тестування, формальна верифікація, перевірка моделі. <i>Самостійна робота: Мови специфікації (універсальні, предметних областей, засобів програмування)</i> | 2 | 4 |
| 3 | Тема 3. Огляд методів верифікації <i>Самостійна робота: Знайомство з можливостями Code Contracts Spec#</i> | 2 | 4 |
| 4 | Тема 4. Тестування, задачі тестування. Основні види тестування, їх характеристика. <i>Самостійна робота: Огляд програмного інструментарію автоматизації тестування</i> | 2 | 4 |
| 5 | Тема 5. Методи доведення коректності програм. <i>Самостійна робота: Доведення коректності програми на заданому прикладі</i> | 2 | 4 |
| 6 | Тема 6. Логіка Гоара для верифікації програм <i>Самостійна робота: VDM, Z</i> | 2 | 4 |
| 7 | Тема 7. Методи Флойда, Маккарті, Дейкстри для верифікації <i>Самостійна робота: Знайомство з можливостями Dafny</i> | 2 | 6 |
| | <i>Контрольна робота 1</i> | | 1 |
| | <i>Захист самостійної роботи</i> | | 1 |
| | Всього по частині 1 | 16 | 30 |
| | Частина 2. Формальні методи верифікації. Логіки програмування. | | |
| 8 | Тема 8. Методи моделювання системи. Поняття специфікації. Модель Кріпке <i>Самостійна робота: Побудова моделі Кріпке відповідно до завдання</i> | 2 | 5 |
| 9 | Тема 9. Лінійна темпоральна логіка (LTL). Синтаксис та семантика LTL. <i>Самостійна робота: Описати твердження в синтаксисі LTL</i> | 2 | 5 |
| 10 | Тема 10. Специфікація властивостей в LTL. Перевірка моделей в LTL. <i>Самостійна робота: Дослідити довільний інструмент перевірки моделі в LTL (на вибір)</i> | 2 | 5 |
| 11 | Тема 11. Верифікація LTL за допомогою автоматів Бюхи <i>Самостійна робота: Дослідити довільний другий інструмент перевірки моделі в LTL (на вибір)</i> | 2 | 5 |
| 12 | Тема 12. Розгалужена темпоральна логіка (CTL). Синтаксис та семантика CTL. <i>Самостійна робота: Описати твердження в синтаксисі CTL. Імовірнісна розгалужена темпоральна логіка.</i> | 2 | 5 |
| 13 | Тема 13. Логіки програмування. <i>Самостійна робота: Знайомство з програмним інструментарієм перевірки моделі в CTL (на вибір)</i> | 2 | 5 |
| | <i>Контрольна робота 2</i> | | 1 |
| | <i>Захист самостійної роботи</i> | | 1 |
| | Всього по частині 2 | 14 | 30 |
| | ВСЬОГО | 30 | 60 |

Загальний обсяг 90 год., в тому числі:

Семінарських занять – 30 год

Самостійна робота - 60 год.

9. Рекомендовані джерела:

Основні

1. Нікітченко М.С. Основи математичної логіки / Нікітченко М.С., Шкільняк С.С. // — ВПЦ Київський ун-т. — К., 2006.
2. Вельдер С.Э. Верификация автоматных программ / Вельдер С.Э., Лукин М.А., Шалыто А.А., Яминов Б.Р. // СПб: Наука, 2011. — 244 с.
3. Жуков Д.Ю. Математическая модель и методы верификации программных систем / Жуков Д.Ю., Миронов А.М. // Информационные технологии и вычислительные системы, 2005. — стр. 49-67.
4. Кларк Э.М. Верификация моделей программ. Model Checking / Кларк Э.М., Грамберг О., Пелед Д. // М.: МЦНМО, 2002. — 416 с.
5. Hoare C.A.R. Axiomatic Bases for Computer Programming // Magazine Communications of the ACM, Vol. 12. — New York, 1969. — Pages 576-580.