

**КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ІМЕНІ ТАРАСА ШЕВЧЕНКА**

ФАКУЛЬТЕТ КОМП'ЮТЕРНИХ НАУК ТА КІБЕРНЕТИКИ

Кафедра теорії та технології програмування

«ЗАТВЕРДЖУЮ»

Заступник декана
з навчальної роботи

_____ Кашпур О.Ф.

«___» _____ 2018 року

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

для студентів

галузь знань **12 «Інформаційні технології»**
(цифр і назва)

спеціальність **122 «Комп'ютерні науки»**
(цифр і назва спеціальності)

освітній рівень **бакалавр**
(молодший бакалавр, бакалавр, магістр)

освітня програма **«Інформатика»**
(назва освітньої програми)

вид дисципліни **вибіркова**

Форма навчання

денна

Навчальний рік

2018/2019

Семестр

8

Кількість кредитів ECTS

3

Мова викладання, навчання
та оцінювання

українська

Форма заключного контролю

залік

Викладачі: **д.ф.-м.н., проф. Нікітченко М.С.** (семінари)

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

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

КИЇВ – 2018

Розробник: Нікітченко Микола Степанович, д.ф.-м.н., професор кафедри «Теорії та технології програмування»

ЗАТВЕРДЖЕНО

В.о. завідувача кафедри теорії та технології
програмування

_____ (Т.В. Панченко)
(підпис) (прізвище та ініціали)

Протокол № __ від «__» _____ 2018 р.

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

Протокол від «__» _____ 2018 року №__

Голова науково-методичної комісії _____ Хусаїнов Д.Я.
(підпис) (прізвище та ініціали)

Схвалено Вченою радою факультету комп'ютерних наук та кібернетики

Протокол від «__» _____ 2018 року №__

Голова Вченої ради _____ Анісімов А.В.
(підпис) (прізвище та ініціали)

Затверджено вченою радою факультету комп'ютерних наук та кібернетики

Протокол від «__» _____ 2018 року №__

Голова вченої ради факультету _____ А.В. Анісімов

ВСТУП

1. Мета дисципліни – засвоєння базових знань з дисципліни «Коректність програм та логіки програмування». Включає в себе 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 синтаксису та семантики, основи функціонального та логічного програмування.
3. *Володіти елементарними навичками:* програмування в сучасних мовах.

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 *спеціалізацією “Теорія та технологія програмування”*.

Викладається в 8 семестрі 4 курсу в обсязі 90 годин.

(3 кредити ECTS)) зокрема: *семінарські заняття – 34 год., самостійна робота – 56 год.* У курсі передбачено 2 частини та 2 контрольні роботи. Завершується дисципліна – **заліком в 8 семестрі.**

В результаті вивчення навчальної дисципліни студент повинен:

знати: основні поняття алгоритму, синтаксису та семантики, основи функціонального та логічного програмування; основи композиційного програмування;

вміти: застосовувати числення предикатів першого порядку, модальні логіки, темпоральні логіки та алгоритмів верифікації.

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:

- здатність до математичного та логічного мислення, формулювання та досліджування математичних моделей, зокрема дискретних математичних моделей, обґрунтовування вибору методів і підходів для розв'язування теоретичних і прикладних задач у галузі комп'ютерних наук, аналізу та інтерпретування;
- здатність формулювати та забезпечувати вимоги щодо якості програмного забезпечення у відповідності з вимогами, технічним завданням та стандартами;
- здатність до алгоритмічного та логічного мислення;
- 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 | | | |
| RH1.1 | Знати необхідні методи верифікації і валідації програмних систем | Семінарські заняття | Контрольна робота 60% правильних відповідей | 20% |
| RH1.2 | Знати логічні засоби специфікації програм та методи перевірки на моделях | Семінарські заняття | Контрольна робота 60% правильних відповідей | 20% |
| RH2.1 | Вміти застосовувати методи доведення правильності програм | Семінарські заняття, самостійна робота | Поточне оцінювання | 20% |
| RH2.2 | Вміти планувати та використовувати методи тестування програмних систем | Семінарські заняття, самостійна робота | Поточне оцінювання | 20% |
| RH3.1 | Обґрунтовувати власний погляд на задачу, спілкуватися з колегами з питань проектування, розробки специфікацій та програм. | Самостійна робота | Поточне оцінювання | 10% |
| RH4.1 | Організовувати свою самостійну роботу для досягнення результату. | Залік, самостійна робота | Поточне оцінювання | 10% |

6. Співвідношення результатів навчання дисципліни із програмними результатами навчання

| Результати навчання дисципліни | RH 1.1 | RH 1.2 | RH 2.1 | RH 2.2 | RH 3.1 | RH 4.1 |
|--|--------|--------|--------|--------|--------|--------|
| Програмні результати навчання | | | | | | |
| (з опису освітньої програми) | | | | | | |
| ПРН12. Вміти застосовувати методи та алгоритми інтелектуального аналізу даних для задач класифікації, прогнозування, кластерного аналізу, пошуку асоціативних правил з використанням програмних інструментів підтримки багатовимірного аналізу даних на основі використання технологій DataMining, TextMining, WebMining. | + | + | + | + | | |
| ПРН18.2. Аналізувати, оцінювати і вибирати інструментальні та обчислювальні засоби, парадигми, технології, алгоритмічні і програмні рішення при проектуванні та розробці програмних систем. | | | + | + | + | + |

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.

У випадку відсутності студента з поважних причин відпрацювання та перездачі контрольних робіт здійснюються у відповідності до „Положення про порядок оцінювання знань студентів при кредитно-модульній системі організації навчального процесу” від 1 жовтня 2010 року.

7.3 Шкала відповідності оцінок

| | |
|----------------------------------|--------|
| Відмінно / Excellent | 90-100 |
| Добре / Good | 75-89 |
| Задовільно / Satisfactory | 60-74 |
| Незадовільно / Fail | 0-59 |
| Зараховано / Passed | 60-100 |
| Не зараховано / Fail | 0-59 |

8. Структура навчальної дисципліни. Тематичний план лекцій

| № лекції | Назва лекції | Кількість годин | | |
|----------|---|-----------------|--------------|-----------|
| | | Лекції | Семін занять | Сам. р-та |
| | Частина 1. Валідація систем | | | |
| 1 | Тема 1. Поняття та задачі валідації. Характеристика основних методів валідації систем: тестування, формальна верифікація, перевірка моделі | | 2 | 4 |
| 2 | Тема 2. Тестування, задачі тестування. Основні види тестування, їх характеристика. | | 2 | 4 |
| 3 | Тема 3. Формальні методи верифікації | | 2 | 4 |
| 4 | Тема 4. Трійки Хоара. | | 2 | 4 |
| 5 | Тема 5. Формальні методи верифікації паралельних систем. | | 2 | 4 |
| 6 | Тема 6. Темпоральна логіка. Перевірка моделі (Model Checking). | | 2 | 4 |
| 7 | Тема 7. Методи Model Checking, переваги та недоліки. Застосування Model Checking. | | 4 | 4 |
| | <i>Контрольна робота 1</i> | | 2 | |
| | Всього по частині 1 | | 16 | 28 |
| | Частина 2. Формальні методи верифікації | | | |
| 8 | Тема 8. Методи моделювання системи. Поняття специфікації. Модель Кріпке | | 2 | 2 |
| 9 | Тема 9. Лінійна темпоральна логіка (LTL). Синтаксис та семантика LTL. | | 2 | 2 |
| 10 | Тема 10. Model Checking, виконуваність, тавтологічність LTL. | | 2 | 4 |
| 11 | Тема 11. Специфікація властивостей в LTL. Перевірка моделей в LTL. | | 2 | 4 |
| 12 | Тема 12. Верифікація LTL за допомогою автоматів Бюхи | | 2 | 4 |
| 13 | Тема 13. Розгалужена темпоральна логіка (CTL). Синтаксис та семантика CTL. | | 2 | 4 |
| 14 | Тема 14. Порівняння CTL та LTL. | | 2 | 4 |
| 15 | Тема 15. Специфікація властивостей в CTL. Перевірка моделей для CTL | | 2 | 4 |
| | <i>Контрольна робота 2</i> | | 2 | |
| | Всього по частині 2 | | 18 | 28 |
| | ВСЬОГО | | 34 | 56 |

Загальний обсяг 90 год., в тому числі:

Семінарських занять – 34 год

Самостійна робота - 56 год.

Теми, винесені на самостійне вивчення:

1. Ймовірнісна розгалужена темпоральна логіка
2. Перевірка моделей для CTL-формул за допомогою верифікатора SML.

9. Рекомендовані джерела:

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.