

**КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ІМЕНІ ТАРАСА ШЕВЧЕНКА**

ФАКУЛЬТЕТ КОМП'ЮТЕРНИХ НАУК ТА КІБЕРНЕТИКИ

Кафедра інформаційних систем

«ЗАТВЕРДЖУЮ»

Заступник декана
з навчальної роботи

_____ Кашпур О.Ф.

«___» _____ 2018 року

**РОБОЧА ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ
ПРОГРАМУВАННЯ**

для студентів

галузь знань **12 «Інформаційні технології»**
(шифр і назва)
спеціальність **121 «Інженерія програмного забезпечення»**
(шифр і назва спеціальності)
освітній рівень **бакалавр**
(молодший бакалавр, бакалавр, магістр)
освітня програма **«Програмна інженерія»**
(назва освітньої програми)

вид дисципліни **обов'язкова**

Форма навчання	денна
Навчальний рік	2018/2019
Семестр	2
Кількість кредитів ECTS	6
Мова викладання, навчання та оцінювання	українська
Форма заключного контролю	екзамен

Викладачі: **к.ф.-м.н., доц. Іванов Є.О.** (лекції, лабораторні заняття),
к.ф.-м.н., доц. Верес М.М. (лабораторні заняття),
к.ф.-м.н., доц. Жереб К.А. (лабораторні заняття)

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

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

КИЇВ – 2018

Розробник: Іванов Євгеній Олександрович, к.ф.-м.н., доцент кафедри «Інформаційних систем»

ЗАТВЕРДЖЕНО

В.о. зав. кафедри «Інформаційних систем»

_____ (Іванов Є.О.)
(підпис) (прізвище та ініціали)

Протокол № 8 від «22» травня 2018 р.

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

Протокол від «18» червня 2018 року №__

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

«_____» _____ 20__ року

1. Мета дисципліни – засвоєння основних понять програмування, опанування методів та засобів, необхідних для представлення складних даних та конструювання ефективних програм роботи з ними. Оволодіння навичками використання мови програмування C++ та відповідних середовищ програмування.

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

1. *Знати:* основні поняття програмування та принципи розробки програм, базові класичні алгоритми та різновиди структур даних.
2. *Вміти:* проектувати, розробляти та тестувати програми на базовому рівні.
3. *Володіти елементарними навичками:* програмування мовами C, C++.

3. Анотація навчальної дисципліни:

Навчальна дисципліна “Програмування” є складовою освітньо-професійної програми підготовки фахівців за першим (бакалаврським) рівнем вищої освіти галузі знань 12 „Інформаційні технології” зі спеціальності 121 „Інженерія програмного забезпечення”, освітньо-професійної програми „Програмна інженерія”.

Дана дисципліна є обов’язковою навчальною дисципліною за *програмою “Програмна інженерія”*.

Викладається у 2 семестрі 1 курсу в **обсязі – 180 год.**

(**6 кредитів ECTS**) зокрема: *лекції – 32 год., лабораторні – 56 год., консультації – 2 год., самостійна робота – 90 год.* У курсі передбачено **2 змістових модулів та 2 модульні контрольні роботи.** Завершується дисципліна – **екзаменом у 2 семестрі.**

В результаті вивчення навчальної дисципліни студент повинен:

знати основні принципи програмування, способи представлення та обробки внутрішньо складних даних, технології проектування, налагодження та тестування програм.

вміти ефективно використовувати можливості мови C++ та відповідних середовищ програмування для створення програмних проектів з відносно складною внутрішньою структурою.

Для допуску до дисципліни „Програмування” освітньо-професійної програми «Програмна інженерія» студент повинен опанувати компетентності та результати навчання, які надає дисципліна „Основи програмування” програми «Програмна інженерія». Дисципліна „Програмування” є базовою для засвоєння дисципліни «Об’єктно-орієнтоване програмування», дисциплін спеціалізації та дисциплін вільного вибору студента програмістського спрямування програми «Програмна інженерія».

4. Завдання (навчальні цілі):

набуття знань, умінь та навичок (компетентностей) на рівні існуючих підходів у програмуванні, відповідно до кваліфікації фахівців з інформаційних технологій. Зокрема, розвивати:

- здатність застосовувати знання у практичних ситуаціях;
- здатність оцінювати та забезпечувати якість виконуваних робіт;
- здатність проектувати та розробляти програмне забезпечення з відносно складною внутрішньою структурою;
- здатність до алгоритмічного мислення.

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 основні поняття та принципи структурного програмування	Лекція, лабораторне заняття	Тест, 60% правильних відповідей, екзамен	15%
РН1.2	Знати способи представлення та обробки внутрішньо складних даних, технології проектування, налагодження та тестування програм	Лекція, лабораторне заняття	Тест, 60% правильних відповідей, екзамен	20%
РН1.3	Знати основні етапи життєвого циклу програм	Лекція, лабораторне заняття	Тест, 60% правильних відповідей, екзамен	15%
РН2.1	Вміти застосовувати на практиці інструментальні програмні засоби проектування та розробки програмного забезпечення.	Лабораторне заняття, самостійна робота	Захист лабораторної роботи, екзамен	24%
РН3.1	Обґрунтовувати власний погляд на задачу, спілкуватися з колегами з питань проектування та розробки програм, складати письмові звіти	Лабораторне заняття	Поточне оцінювання, захист ЛР	10%
РН4.1	Організувати свою самостійну роботу для досягнення результату	Самостійна робота	Поточне оцінювання, Захист лабораторної роботи	8%
РН4.2	Відповідально ставитися до виконуваних робіт, нести відповідальність за їх якість	Лабораторна робота	Захист лабораторної роботи	8%

6. Співвідношення результатів навчання дисципліни із програмними результатами навчання

Результати навчання дисципліни	РН 1.1	РН 1.2	РН 1.3	РН 2.1	РН 3.1	РН 4.1	РН 4.2
Програмні результати навчання							
(з опису освітньої програми)							
ПРН-12. Знати ефективні підходи щодо проектування програмного забезпечення.	+		+			+	+
ПРН-28.1. Знати та мати навички реалізації основних алгоритмів та структур даних програмування.		+		+	+		

7. Схема формування оцінки.

7.1 Форми оцінювання студентів:

- семестрове оцінювання:

1. Контрольна робота 1: РН 1.1, РН 1.2 — 10 балів/6 балів.
2. Контрольна робота 2: РН 1.2, РН1.3 — 10 балів/6 балів.
3. Лабораторна робота 1 (проект): РН 1.2, РН1.3, РН 2.1, РН3.1 – 8 балів/4 бали.
4. Лабораторна робота 2 (проект): РН1.2, РН1.3, РН 2.1, РН3.1 – 8 балів/4 бали.
5. Лабораторна робота 3 (проект): РН1.2, РН1.3, РН 2.1, РН3.1 – 8 балів/4 бали.
6. Лабораторна робота 4 (проект): РН1.2, РН1.3, РН 2.1, РН3.1 – 8 балів/4 бали.
7. Лабораторна робота 5 (проект): РН1.2, РН1.3, РН 2.1, РН3.1 – 8 балів/4 бали.

- підсумкове оцінювання (у формі екзамену) вказується:

- максимальна кількість балів які можуть бути отримані студентом: 40 балів;
 - результати навчання які будуть оцінюватись: РН1.1, РН1.2, РН1.3, РН2.1;
 - форма проведення і види завдань: письмова.
- Види завдань: 5 письмових завдань.

Критерії оцінювання на екзамені

Завдання	Тема завдання	Максимальний відсоток від 40 балів	Всього відсотків
Завдання 1-5	Задача на програмування мовою C++	По 20%	100%
			100%

Студент не допускається до екзамену, якщо під час семестру набрав менше ніж 24 балів. Студент допускається до екзамену за умови виконання 70% передбачених планом лабораторних робіт.

7.2 Організація оцінювання:

Терміни проведення форм оцінювання:

1. Контрольна робота : до 7 тижня семестру.
2. Контрольна робота : до 15 тижня семестру.
3. Лабораторна робота 1 (проект): до 4 тижня семестру.
4. Лабораторна робота 2 (проект): до 7 тижня семестру.
5. Лабораторна робота 3 (проект): до 10 тижня семестру.
6. Лабораторна робота 4 (проект): до 13 тижня семестру.
7. Лабораторна робота 5 (проект): до 16 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.

У разі неякісного виконання лабораторної роботи, викладач має право не зарахувати лабораторну роботу, або знизити за неї бали.

Студент має право здавати лабораторні роботи після закінчення визначеного для них терміну, але з втратою одного балу за кожен тиждень, який пройшов з моменту закінчення терміну її здачі.

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 і лабораторних занять

№ лекції	Назва лекції	Кількість годин		
		Лекції	Лабораторні	Самостійна
Частина 1. Лінійні динамічні структури даних. Сортування.				
1.	Тема 1. Динамічні структури даних. Інструментарій.	2	2	4
2.	Тема 2. Списки та рекурсія.	2	4	6
3.	Тема 3. Лінійні списки. Операції. Послідовні та зв'язні способи збереження.	2	4	6
4.	Тема 4. Стеки та черги. Стеки та рекурсія.	2	4	6
5.	Тема 5. Стиснені та індексні способи збереження лінійних списків.	2	4	6
6.	Тема 6. Прості методи сортування.	2	2	4
7.	Тема 7. Швидке сортування. Сортуванням файлів.	2	4	6
8.	Тема 8. Матриці та багатовимірні масиви.	2	2	4
	Контрольна робота 1			
Контроль за підсумками лабораторних робіт 1 та 2				
Всього по частині 1		16	26	42
Частина 2. Застосування нелінійних динамічних структур даних				
9.	Тема 9. Дерева. Представлення та проходження.	2	4	6
10.	Тема 10. Бінарні дерева. Представлення та обробка.	2	4	6
11.	Тема 11. Представлення та обробка виразів.	2	4	6
12.	Тема 12. Графи, способи їх представлення та обробка.	2	4	6
13.	Тема 13. Програмування алгоритмів на графах.	2	4	6
14.	Тема 14. Пошук в лінійних списках. Дерева бінарного пошуку. Оптимальні дерева пошуку.	2	4	6
15.	Тема 15. Схеми збалансованих дерев. AVL - дерева. 2-3 - дерева.	2	4	6
16.	Тема 16. Елементи технології проектування, програмування та тестування.	2	2	6
	Контрольна робота 2			
Контроль за підсумками лабораторних робіт 3, 4 та 5				
Всього по частині 2		16	30	48
Консультація				
Екзамен				
ВСЬОГО		32	56	90

Загальний обсяг 180 год., в тому числі:

Лекцій – 32 год.

Лабораторні заняття - 56 год.

Консультації – 2 год.

Самостійна робота - 90 год.

Теми, винесені на самостійне вивчення:

Програмування алгоритмів сортування та пошуку.
Програмування алгоритмів лінійної алгебри та дискретної математики.
Застосування списків, стеків, черг.
Можливості для стиснення та індексування даних.
Синтаксичний аналіз на обробка виразів.
Алгоритми для задач на графах.
Схеми збалансованих дерев двійкового пошуку.
Елементи технології програмування.
Виконання лабораторних робіт 1-5.

9. Рекомендовані джерела:

Основні

1. Ахо А. , Хопкрофт Дж., Ульман Дж. Структуры данных и алгоритмы. – М.: Издательский дом «Вильямс», 2000.
2. Рейнгольд Э., Нивергельт, Део Н. Комбинаторные алгоритмы. Теория и практика. - М.: Мир, 1980.
3. Прата С. Язык программирования C++. Лекции и упражнения. М.: ООО «И.Д. Вильямс», 2007.
4. Павловская Т.А. C/C++. Программирование на языке высокого уровня. СПб.: Питер, 2003.
5. Дейтел Х., Дейтел П. Как программировать на C++.
6. Керниган Б., Пайк Р. Практика программирования. – СПб.: Невск. Диалект, 2001.
7. Окулов С.М. Программирование в алгоритмах. - М.: Бином. Лаборатория знаний, 2004.
8. Проценко В.С., Чаленко П.И., Сорока Р.А. Техника программирования. - К.: Выща школа, 1990.
9. Ван Тассел Д. Стиль, разработка, эффективность, отладка и испытание программ. - М.: Мир, 1985.

Додаткові:

10. Липпман С., Лажоие Ж. Язык программирования C++. Вводный курс. – СПб.: Невск. Диалект, 2001.
11. Ковалюк Т.В. Основы програмування. – К.: Видавнича група ВНУ, 2005.
12. Страуструп Б. Язык программирования C++. Специальное издание. – СПб.: Невск. Диалект, 2006.
13. Ключин Д.А. Полный курс C++. Профессиональная работа. – М.: Вильямс, 2005.
14. Ставровський А.Б., Карнаух Т.О. Програмування. Перші кроки. М.: Видавничий дім "Вільямс", 2005.
15. Керниган Б., Плоджер Ф. Элементы стиля программирования. М.: Радио и связь, 1984.
16. Керниган Б., Ричи Д. Язык программирования Си. – СПб.: Невск. Диалект, 2001.
17. Коплиен Дж. Программирование на C++. – СПб.: Питер, 2004.
18. Лаптев В.В., Морозов А.В., Бокова А.В. C++. Объектно-ориентированное программирование. Задачи и упражнения. – СПб.: Питер, 2007.
19. Сэдживик Р. Фундаментальные алгоритмы на C++. М.: ООО «ДиаСофтЮП», 2002.
20. Харбисон С.П., Стил Г.Л. Язык программирования С. – М.: БИНОМ, 2004.
21. Шилдт Г. Полный справочник по С. – М.: Вильямс, 2006.
22. Касьянов В.Н., Сабельфельд В.К., Трахтенброт М.Б. Сборник индивидуальных заданий по практике на ЭВМ. - Новосибирск, НГУ, 1981.
23. Касьянов В.Н., Поттосин И.В., Сабельфельд В.К., Трахтенброт М.Б. Индивидуальные задания по практике на ЭВМ. - Новосибирск, НГУ, 1981.
24. Кушниренко А.Г., Лебедев Г.В. Программирование для математиков: Учебн. пособие для вузов. -М.: Наука, 1988.

25. Проценко В.С., Чаленко П.Й., Ставровський А.Б. Техніка програмування мовою Сі. - К.: Либідь, 1993.
26. Ускова О.Ф. и др. Программирование алгоритмов обработки данных. – СПб.: БХВ-Петербург, 2003.
27. Глибовець М.М. Основи комп'ютерних алгоритмів.-К.:Вид. дім «КМ Академія», 2003.
28. Глибовець М.М., Ляшко В.І, Проценко В.С. Мова програмування Сі: навч.-метод. посіб. – К.: ВД «КМ Академія», 2002.
29. Зубенко В.В., Омельчук Л.Л. Програмування: навчальний посібник. – К.: ВПЦ: «Київський університет», 2011.
30. Дж.Макконнел Основы современных алгоритмов.- Техносфера, 2004.
31. Дейкстра Э. Дисциплина программирования. - М.: Мир, 1978.
32. Зелковиц М., Шоу А., Гэннон Дж. Принципы разработки программного обеспечения. – М.: Мир, 1982.
33. Йодан Э. Структурное проектирование и конструирование программ. – М.: Мир, 1979.
34. Гэри М., Д. Джонсон. Вычислительные машины и труднорешаемые задачи. - М.: Мир, 1982.
35. Себеста Р. Основные концепции языков программирования. – М.: Издательский дом «Вильямс», 2001.
36. Уэзерелл Ч. Этюды для программистов. - М.: Мир, 1982.
37. Майерс Г. Искусство тестирования программ. - М.: Радио и связь, 1982.
38. Майерс Г. Надежность программного обеспечения. - М.: Мир, 1980.
39. Мейер Б., Бодуэн К. Методы программирования. Т.1, Т.2. - М.: Мир, 1982.
40. Ахо А. , Хопкрофт Дж., Ульман Дж. Построение и анализ вычислительных алгоритмов. – М.: Мир, 1979.
41. Липский В. Комбинаторика для программистов. - М.: Мир, 1988.
42. Евстигнеев В.А. Применение теории графов в программировании. /Под ред. А.П. Ершова. - М.: Наука, 1985.
43. Гудман С., Хидетниемеи С. Введение в разработку и анализ алгоритмов. - М.: Мир, 1981.
44. Кнут Д. Искусство программирования. Т.1: Основные алгоритмы. - М.: Мир, 1976.
45. Кнут Д. Искусство программирования. Т.2: Получисленные алгоритмы. - М.: Мир, 1976.
46. Кнут Д. Искусство программирования. Т.3: Сортировка и поиск. - М.: Мир, 1976.
47. Порублев И.Н., Ставровский А.Б. Алгоритмы и программы. Решение олимпиадных задач.- М.:ООО «И.Д.Вильямс», 2007.
48. Иванов Б.Н. Дискретная математика. Алгоритмы и программы. - М.: Лаборатория Базовых Знаний, 2001.
49. Новиков Ф.А. Дискретная математика для программистов. - СПб.: Питер, 2001.
50. Буч Г. Объектно-ориентированное проектирование с примерами применения. – М.: Конкорд, 1992.