

**КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ІМЕНІ ТАРАСА ШЕВЧЕНКА**

ФАКУЛЬТЕТ КОМП'ЮТЕРНИХ НАУК ТА КІБЕРНЕТИКИ

Кафедра теоретичної кібернетики

«ЗАТВЕРДЖУЮ»

Заступник декана
з навчальної роботи

_____ Кашпур О. Ф.
«__» _____ 2018 року

**РОБОЧА ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ
ПРОГРАМУВАННЯ**

для студентів

галузь знань **12 «Інформаційні технології»**
спеціальність **122 «Комп'ютерні науки»**
освітній рівень **бакалавр**
освітня програма **«Інформатика»**

вид дисципліни **обов'язкова**

Форма навчання	денна
Навчальний рік	2018/2019
Семестр	1, 2
Кількість кредитів ECTS	10
3 них семестр 1	5
семестр 2	5
Мова викладання, навчання та оцінювання	українська
Форма заключного контролю	екзамен

Викладачі: **к.ф.-м.н, доц. Ставровський А. Б.** (лекції, лабораторні заняття),
асистент Коваль Ю. В. (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: Ставровський Андрій Борисович, канд. фіз.-мат. н., доцент,
доцент кафедри теоретичної кібернетики

ЗАТВЕРДЖЕНО
Зав. кафедри теоретичної кібернетики

_____ (Крак Ю.В.)
(підпис) (прізвище та ініціали)

Протокол № __ від «__» _____ 20__ р.

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

Протокол від «__» _____ 20__ року №__

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

«__» _____ 20__ року

1 Мета дисципліни – знайомство з початками програмування, базовою термінологією, засобами керування порядком обчислень у мові програмування високого рівня, класичними задачами, оволодіння елементами технології створення програм.

2 Попередні вимоги до опанування або вибору навчальної дисципліни

Знати: математику та інформатику в об'ємі шкільного курсу

Вміти: застосовувати знання з природничих дисциплін шкільного курсу до розв'язання задач

Володіти елементарними навичками: роботи з комп'ютером

3 Анотація навчальної дисципліни

Дисципліна є складовою ОПП підготовки фахівців за першим (*бакалаврським*) рівнем вищої освіти *галузі знань* 12 «Інформаційні технології» зі *спеціальності* 122 «Комп'ютерні науки», *освітньо-професійної програми* «Інформатика». Дисципліна є обов'язковою навчальною дисципліною. Викладається в 1-му й 2-му семестрах, обсяг 300 год. (10 кредитів ECTS), з них лекції – 68 год., лабораторні – 72 год., консультації – 4 год., самостійна робота – 156 год. Передбачено 4 змістових модулі, 4 модульні контрольні роботи та екзамени в обох семестрах.

В результаті вивчення навчальної дисципліни студент повинен:

знати зображення числових та інших даних, засоби керування порядком обчислень, організацію структурних даних, класи та інкапсуляцію, використання потоків, успадкування класів, зображення графів у програмах, складність алгоритмів і проблем.

вміти застосовувати на практиці інструментальні засоби програмування, розв'язувати навчальні та практичні задачі, обґрунтовувати власний погляд на розв'язання задачі, спілкуватися з колегами з питань програмування, складати звіти з розв'язання задач.

Дисципліна використовує поняття з дискретної математики та алгебри. Виступає базовою для дисциплін: «Об'єктно-орієнтоване програмування», «Алгоритміка», «Бази даних та інформаційні системи», «Інструментальні середовища та технології програмування», «Системне програмування», «Теорія програмування».

4 Завдання (навчальні цілі)

Набуття базових знань, умінь та навичок (компетентностей) з програмування відповідно до освітньої кваліфікації «Бакалавр з комп'ютерних наук». Зокрема, розвивати:

- здатність застосовувати знання в практичних ситуаціях,
- здатність вчитися й оволодівати сучасними знаннями,
- здатність оцінювати та забезпечувати якість виконуваних робіт,
- здатність до побудови логічних висновків, використання формальних мов і моделей алгоритмічних обчислень, проектування, розроблення й аналізу алгоритмів.

5 Результати навчання за дисципліною

Результат навчання (РН) (1 – знати; 2 – вміти; 3 – комунікація; 4 – автономність та відповідальність)		Форми викладання та навчання	Методи оцінювання	Відсоток у підсумковій оцінці з дисципліни
Код	Результат навчання			
РН1.1	Знати основні поняття програмування та принципи розробки програм	Лекція, лабораторне заняття	Модульна контрольна робота (МКР), екзамен	40
РН1.2	Знати базові класичні алгоритми та різновиди структур даних	Лекція, лабораторне заняття		
РН2.1	Вміти проектувати, розробляти та тестувати програми	Лабораторна робота, самостійна робота	МКР, здача ЛР, екзамен	30
РН3.1	Обґрунтовувати власний погляд на задачу, спілкуватися з колегами з питань проектування та розробки	Лабораторне заняття	Поточне оцінювання (ПО), здача	10

	програм, складати письмові звіти		ЛР	
РН4.1	Організувати свою самостійну роботу для досягнення результату	Самостійна робота	ПО, здача ЛР	10
РН4.2	Відповідально ставитися до виконуваних робіт, нести відповідальність за їх якість	Лабораторна робота	Здача ЛР	10

6 Співвідношення результатів навчання дисципліни з програмними результатами навчання

Програмні результати навчання <i>(з опису освітньої програми)</i>	Результати навчання дисципліни					
	РН1.1	РН1.2	РН2.1	РН3.1	РН4.1	РН4.2
ПР4. Проектувати, розробляти та аналізувати алгоритми розв'язання обчислювальних та логічних задач, оцінювати ефективність та складність алгоритмів на основі застосування формальних моделей алгоритмів та обчислюваних функцій.	+	+	+	+		
ПР11. Володіти навичками використання методології управління життєвим циклом програмного забезпечення, продуктів і сервісів інформаційних технологій відповідно до вимог і обмежень замовника, вміти готувати проектну документацію.					+	+

7 Схема формування оцінки

7.1 Форми оцінювання студентів:

- семестрове оцінювання:

Перший семестр

1. Модульна контрольна робота 1: РН1.1 – 20 б./9 б.
2. Модульна контрольна робота 2: РН1.1, РН2.1 – 20 б./9 б.
3. Лабораторна робота 1: РН3.1, РН4.1, РН4.2 – 10 б./6 б.
4. Лабораторна робота 2: РН1.1, РН3.1, РН4.1, РН4.2 – 10 б./6 б.

Другий семестр

6. Модульна контрольна робота 3: РН1.2, РН2.1 – 20 б./9 б.
7. Модульна контрольна робота 4: – 20 б./9 б.
8. Лабораторна робота 4: РН1.2, РН3.1, РН4.1, РН4.2 – 4 б./3 б.
9. Лабораторна робота 5: РН1.2, РН3.1, РН4.1, РН4.2 – 4 б./3 б.
10. Лабораторна робота 6: РН1.2, РН3.1, РН4.1, РН4.2 – 4 б./3 б.
10. Лабораторна робота 7: РН1.2, РН3.1, РН4.1, РН4.2 – 4 б./3 б.
10. Лабораторна робота 8: РН1.2, РН3.1, РН4.1, РН4.2 – 4 б./3 б.

- підсумкове оцінювання (у формі екзамену в кожному семестрі):

- максимальна кількість балів які можуть бути отримані студентом: 40;
- результати навчання, які оцінюються: РН1.1, РН1.2, РН2.1, РН3.1;
- форма проведення: письмова
- види завдань: п'ять теоретичних питань (40%), дві задачі (60%).

Студент допускається до екзамену, якщо виконав не менше 70% лабораторних робіт, передбачених планом, і в семестрі набрав не менше ніж 20 балів. Для отримання загальної позитивної оцінки з дисципліни оцінка за екзамен має бути не менше 24 балів.

7.2 Організація оцінювання

Терміни проведення оцінювання в першому семестрі

1. Модульні контрольні роботи: № 1 – до 7 тижня, № 2 – до 13 тижня.

2. Лабораторні роботи: № 1 – до 6 тижня, № 2 – до 10 тижня, № 3 – до 14 тижня.

Терміни проведення оцінювання в другому семестрі

1. Модульні контрольні роботи: № 1 – до 7 тижня, № 2 – до 13 тижня.

2. Лабораторні роботи: № 1 – до 6 тижня, № 2 – до 10 тижня, № 3 – до 14 тижня.

Студент має право один раз перескласти модульну контрольну роботу з можливістю отримати не більше 80% балів, призначених за роботу. Термін перескладання визначає викладач.

За відсутності студента з поважних причин перездача МКР здійснюється відповідно 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 10% балів за кожен тиждень, що пройшов від 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

8. Структура навчальної дисципліни. Тематичний план лекцій і лабораторних занять

Перший семестр

№ п/п	Назва теми	Кількість годин		
		Лекції	Лабораторні	Самостійна робота
Змістовий модуль 1. Основи організації обчислень				
1.	Тема 1. Базові поняття програмування. Основи мови програмування C++.	2	2	4
2.	Тема 2. Типи даних. Константи. Операції.	2	2	4
3.	Тема 3. Змінні. Арифметичні операції та вирази.	2	2	4
4.	Тема 4. Порівняння та булеві операції.	2	2	4
5.	Тема 5. Інструкції розгалуження та вибору варіантів.	2	2	4
6.	Тема 6. Функції. Параметри та аргументи.	2	2	4
7.	Тема 7. Інструкції циклу. Переривання циклу.	2	2	4
8.	Тема 8. Рекурентні співвідношення та цикли.	2	2	4
9.	Тема 9. Рекурсія та рекурсивні функції.	2	2	6
10.	Тема 10. Особливі ситуації та винятки.	2	2	6
11.	Тема 11. Обробка та отримання послідовностей.	2	2	6
12.	Тема 12. Структурна організація програми.	2	2	6
	<i>Модульна контрольна робота 1</i>			
	Всього по модулю 1	24	24	56
Змістовий модуль 2. Масиви та структури				
1.	Тема 13. Масиви, адреси та вказівники.	2	2	4
2.	Тема 14. Масиви у функціях.	2	2	4
3.	Тема 15. Багатомірні масиви.	2	2	4
4.	Тема 16. Рядки мови C. Параметри головної функції.	2	2	6
5.	Тема 17. Структури.	2	2	6
	<i>Модульна контрольна робота 2</i>			
	Всього по модулю 1	10	10	24

Загальний обсяг **150** год. (5 кредитів ECTS), у тому числі:

Лекцій – **34** год.

Лабораторні – **34** год.

Консультації – 2 год.

Самостійна робота – 80 год.

Типові завдання модульних контрольних робіт

Модульна контрольна робота № 1

1. Дати означення кількох базових понять програмування та мови C++.
2. Записати задане число в заданій системі числення.
3. Записати додатковий код заданого цілого від'ємного числа.
4. Описати обчислення заданого виразу.
5. Написати вираз-ознаку певної властивості.
6. Написати програму з інструкціями розгалуження для обчислень за формулою з набором умов.

Модульна контрольна робота № 2

1. Означити кілька понять, зв'язаних з циклами, рекурсією, масивами та вказівниками.
2. Написати систему рекурентних співвідношень для обчислення величини за формулою.
3. Написати функцію циклічного обчислення величини за рекурентними співвідношеннями.
4. Імітувати програму з рекурсивною функцією.
5. Імітувати програму обробки масиву.

Питання на іспит

1. Базові поняття програмування.
2. Позиційні системи числення, форми зображення цілих чисел.
3. Принципи зображення дійсних чисел.
4. Алгоритми перетворення записів чисел у різних системах.
5. Лексика мови C++, необхідні елементи програми.
6. Оголошення імен змінних, запис констант, запис арифметичних виразів.
7. Компіляція, компонування й запуск програми.
8. Бібліотечні математичні функції.
9. Запис виразів з операторами порівняння та булевими операторами.
10. Семантика функції та її виклику, параметри та аргументи.
11. Цикл, умова продовження циклу, вигляд і семантика інструкцій циклу.
12. Структурованість інструкцій, переривання циклу.
13. Рекурсивні означення, рекурсивні функції.
14. Глибина рекурсії та її вплив на розміри програмного стеку
15. Загальна кількість рекурсивних викликів та її вплив на час виконання програми.
16. Одиниця трансляції, означення та оголошення імен змінних, простори імен.
17. Масив, елемент масиву.
18. Адреса та адресний вираз, вказівник.
19. Вказівники та індекси елементів масиву.
20. Параметри, що зображують масиви у функціях.
21. Масив, елементами якого є масиви.
22. Багатовимірні масиви як параметри функцій.
23. Рядки мови C, бібліотечні функції їх обробки.
24. Поняття структури, елемент структури та його позначення.
25. Оголошення типу структур.

Другий семестр

№ п/п	Назва теми	Кількість годин		
		Лекції	Лабораторні	Самостійна робота
Змістовий модуль 3. Класи, динамічні дані, успадкування класів				
1.	Тема 18. Класи та об'єкти. Інкапсуляція. Статичні атрибути.	2	2	4
2.	Тема 19. Конструктори та деструктор.	2	2	4
3.	Тема 20. Стандартні класи потоків. Робота з потоками.	2	2	4
4.	Тема 21. Поняття АД. АД «стек». АД «черга».	2	2	4

5.	Тема 22. Динамічні дані. Реалізація стеків і черг у динамічних масивах.	2	2	4
6.	Тема 23. Зв'язані динамічні структури. Зв'язані списки.	2	2	4
7.	Тема 24. Стандартний клас рядків.	2	2	4
8.	Тема 25. Відкрите успадкування. Ієрархії класів.	2	2	4
9.	Тема 26. Захищене та приховане успадкування.	2	2	4
10.	Тема 27. Віртуальні методи та поліморфізм.	2	2	4
11.	Тема 28. Суто віртуальні методи та абстрактні класи.	2	2	6
	<i>Модульна контрольна робота 3</i>			
	Всього по модулю 3	22	22	46
Змістовий модуль 4. Класи графів та супутні питання				
12.	Тема 29. Сім'я класів для зображення графів. Реалізація класів для зображення графів.	2	2	4
13.	Тема 30. Алгоритми обходу графа вглиб, їх реалізація.	2	2	4
14.	Тема 31. Алгоритми обходу графа вшир, їх реалізація.	2	2	4
15.	Тема 32. Алгоритми обходу в розв'язанні задач.	2	2	6
16.	Тема 33. Складність алгоритму. Оцінки складності.	2	4	6
17.	Тема 34. Складність алгоритмів обробки графів.	2	4	6
	<i>Модульна контрольна робота 4</i>			
	Всього по модулю 1	12	16	30

Загальний обсяг **150** год. (5 кредитів ECTS), у тому числі:

Лекцій – **34** год.

Консультації – **2** год.

Лабораторні – **38** год.

Самостійна робота – **76** год.

Типові завдання модульних контрольних робіт

Модульна контрольна робота № 3

1. Означити кілька термінів, зв'язаних з класами, абстрактними типами даних, динамічними даними.
2. Означити типи структур та класів.
3. Написати функцію обробки числового масиву.
4. Означити структури та класи, об'єкти яких містять вказівники на динамічні структури.

Модульна контрольна робота № 4

1. Означити кілька термінів, зв'язаних з успадкуванням класів.
2. Написати класи для роботи з послідовностями зв'язаних динамічних структур.
3. Реалізувати операції з послідовностями зв'язаних структур.
4. Написати абстрактний клас та ієрархію похідних класів для роботи з графами.
5. Реалізувати віртуальні функції в похідних класах.

Питання на іспит

1. Клас, об'єкт класу, атрибути й методи класу, інтерфейс класу.
2. Інкапсуляція та переваги, які вона дає.
3. Конструктор: призначення та ситуації, в яких він викликається.
4. Конструктор копії, його синтаксичні особливості. Ситуації, в яких він викликається.
5. Деструктор: призначення та ситуації, в яких він викликається.
6. Основні частини пам'яті програми, вільна пам'ять.
7. Операції мови C++ створення та знищення динамічних змінних.
8. Необхідність власних конструкторів і деструкторів об'єктів із вказівниками.
9. Глибоке копіювання та його необхідність для обробки об'єктів із вказівниками.
10. Створення й знищення динамічного масиву за допомогою операцій.
11. Абстрактний тип даних і його реалізація.
12. Абстрактний тип даних «стек» і можливі структури даних для зображення стеків.
13. Абстрактний тип даних «черга» і можливі структури даних для зображення черг.

14. Основні переваги класу string порівняно з масивами символів.
15. Параметри головної функції та їх позначення.
16. Успадкування класів, специфікатор доступу в оголошенні успадкування.
17. Сумісність за присвоюванням між похідними та базовими класами.
18. Чим небезпечно встановлення вказівника на базовий клас на об'єкт похідного класу
19. Чим небезпечно встановлення вказівника на похідний клас на об'єкт базового класу
20. Порядок викликів конструкторів і деструкторів для об'єкта похідного класу.
21. Віртуальний метод, синтаксичні й семантичні особливості.
22. Поняття поліморфізму, що дає поліморфізм.
23. Різновиди поліморфізму в C++ і засоби їх реалізації; поліморфний клас.
24. Суто віртуальна функція та її синтаксичні особливості.
25. Абстрактний клас та його головне призначення.
26. Зв'язування виклику функції з її кодом, статичне та динамічне зв'язування.
27. Зображення матриці суміжності графа за допомогою масивів.
28. Зображення структури суміжності графа зв'язаними структурами.
29. Список ребер незваженого графа й способи його зображення.
30. Ефективність виконання типових операцій з графами залежно від їх зображення.
31. Алгоритм обходу вглиб і алгоритм, що його використовує.
32. Розмір екземпляра вхідних даних.
33. Часова складність алгоритму.
34. Оцінки функції згори, знизу, точна.
35. Порядки зростання функцій.
36. Часова складність задачі.

9. Рекомендовані джерела

Основна

1. Белов Ю. А. Вступ до програмування мовою C++. Організація обчислень. / Ю.А. Белов, Т.О. Карнаух, Ю.В. Коваль, А.Б. Ставровський. – К.: Видавничо-поліграфічний центр «Київський університет», 2012. – 175 с.
2. Карнаух Т. О. Вступ до програмування мовою C++. Організація даних./ Т. О. Карнаух, Ю. В. Коваль, М. В. Потієнко, А. Б. Ставровський. – К.: Видавничо-поліграфічний центр «Київський університет», 2015. – 156 с.
3. Либерти Дж. Освой самостоятельно C++ за 21 день. 5-е издание. / Либерти Дж., Брэдли Дж. – М.: Вильямс, 2010. – 784 с.
4. Страуструп Б. Программирование с примерами на C++: принципы и практика. / Страуструп Б. – М.: Вильямс, 2010. – 1084 с.
5. Кормен Т. Алгоритмы. Построение и анализ. / Кормен Т., Лейзерсон Ч., Ривест Р., Штайн К. – М.: Вильямс, 2005. – 1296 с.

Додаткова

6. Ахо А. Структуры данных и алгоритмы. / Ахо А., Дж. Хопкрофт, Дж. Ульман. – М.: Вильямс, 2002. – 384 с.
7. Керниган Б. Практика программирования. / Керниган Б., Ритчи Р. – М.: Вильямс, 2004. – 448 с.
8. Холзнер С. Visual C++ 6. Учебный курс. / Холзнер С. – СПб.: Питер, 2008. – 570 с.
9. Дэвис С. C++ для “чайников”. / Дэвис С. – М.: Изд. дом Вильямс, 2010. – 336 с.
10. Седжвик Р. Алгоритмы на C++. / Седжвик Р. – М.: Вильямс, 2010. – 1056 с.
11. Мюссер Д. C++ и STL: справочное руководство. / Мюссер Д., Дердж Ж., Сейни А.М. – Вильямс, 2010. – 432 с.
12. Шилдт Г. Полный справочник по C++. / Шилдт Г. – М.: Вильямс, 2010. – 800 с.