

КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ІМЕНІ ТАРАСА ШЕВЧЕНКА

ФАКУЛЬТЕТ КОМП'ЮТЕРНИХ НАУК ТА КІБЕРНЕТИКИ

Кафедра теоретичної кібернетики

«ЗАТВЕРДЖУЮ»
Заступник декана
з навчальної роботи



Кашпур О.Ф.

« 28 » серпня 2020 року

РОБОЧА ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ
Структури даних і алгоритми

для студентів

галузь знань	12 «Інформаційні технології» <i>(шифр і назва)</i>
спеціальність	122 «Комп'ютерні науки» <i>(шифр і назва спеціальності)</i>
освітній рівень	бакалавр <i>(молодший бакалавр, бакалавр, магістр)</i>
освітня програма	«Інформатика» <i>(назва освітньої програми)</i>
Вибірковий блок	«Інформаційні технології та системи» <i>(назва спеціалізації)</i>
вид дисципліни	вибіркова

Форма навчання	заочна
Навчальний рік	2020/2021
Семестр	7
Кількість кредитів ECTS	4
Мова викладання, навчання та оцінювання	українська
Форма заключного контролю	іспит

Пролонговано: на 2021/2022 н.р. _____



на 20__/20__ н.р. _____

КИЇВ – 2020

Розробник: Анатолій ПАШКО, д.ф.-м.н., професор, професор кафедри теоретичної кібернетики


ЗАТВЕРДЖЕНО

Зав. кафедри теоретичної кібернетики


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

Протокол № 1 від «27» 08 2020 р.

Схвалено Гарантом освітньо-професійної програми «Інформатика»


_____ Омельчук Л.Л.
(підпис)

«28» 08 2020 р.

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

Протокол від «28» 08 2020 року № 1

Голова науково-методичної комісії _____ Омельчук Л.Л. К
(підпис)

«28» 08 2020 року

1. Мета дисципліни – формування системи знань в області алгоритмізації та структур даних, вмінь і навичок складання алгоритмів та вибору типів структур, необхідних для створення програмного забезпечення. Оволодіння технікою розробки програмного забезпечення для реалізації сучасних алгоритмів в задачах штучного інтелекту.

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

1. *Знати* парадигми програмування, дискретну математику, математичну логіку, теорію алгоритмів та основи програмування в об'ємі стандартних університетських курсів.
2. *Вміти* застосовувати знання з вказаних вище дисциплін до розв'язання задач.
3. *Володіти елементарними навичками* використання логіко-математичної символіки та основами програмування .

3. Анотація навчальної дисципліни:

Навчальна дисципліна «Структури даних і алгоритми» розглядає базові структури даних і основні обчислювальні алгоритми, дозволяє придбати практичні навички з проектування, розробки та аналізу алгоритмів при розв'язуванні прикладних задач та при розробці програмного забезпечення.

Отримані знання дозволять ефективно застосовувати на практиці методи та алгоритми для розв'язування навчальних та практичних задач, обґрунтовувати власний погляд на розв'язання задачі, спілкуватися з колегами з питань програмування, складати звіти з розв'язання задач.

Викладається у 6 семестрі **3** курсу в **обсязі – 120 год. (4 кредити ECTS)**, зокрема: *лекції – 7 год., консультації – 1 год., лабораторні – 3 год., самостійна робота – 109 год.*

4. Завдання (навчальні цілі):

Набуття поглиблених компетентностей алгоритмічного та логічного мислення. Зокрема, розвивати:

СК3. Здатність до побудови логічних висновків, використання формальних мов і моделей алгоритмічних обчислень, проектування, розроблення й аналізу алгоритмів, оцінювання їх ефективності та складності, розв'язності та нерозв'язності алгоритмічних проблем для адекватного моделювання предметних областей і створення програмних та інформаційних систем.

ВСК19.3. Здатність застосовувати математичний апарат та принципи програмування в процесі розробки програмних систем..

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 методи та алгоритми, їх реалізацію сучасними мовами програмування	Лекція, самостійна робота (СР)	контрольна робота (КР), іспит	20
РН1.2	Знати принципи застосування основних сучасних алгоритмів при розробці елементів програмного забезпечення	Лекція, СР	контрольна робота (КР), іспит	20
РН2.1	Вміти застосовувати сучасні алгоритми при розробці елементів програмного забезпечення	Лекція, СР	КР,	20

РН3.1	Обґрунтовувати власний погляд на задачу, спілкуватися з колегами з питань проектування та розробки програм, складати письмові звіти	Лекція, СР	КР	10
РН4.1	Організувати свою самостійну роботу для досягнення результату	Самостійна робота	КР	10
РН4.2	Відповідально ставитися до виконуваних робіт, нести відповідальність за їх якість	Лекція, СР	КР, іспит	20

6 Співвідношення результатів навчання дисципліни з програмними результатами навчання

Результати навчання дисципліни Програмні результати навчання <i>(з опису освітньої програми)</i>	РН1.1	РН1.2	РН2.1	РН3.1	РН4.1	РН4.2
	ПРН4. 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.	+	+	+	+	
ПРН17.3. Знати математичний апарат та принципи програмування та вміти застосовувати їх у створенні програмних систем.		+	+	+	+	
ПРН18.3. Знати фази та ітерації життєвого циклу програмних систем.	+				+	+

7. Схема формування оцінки.

7.1 Форми оцінювання студентів:

- семестрове оцінювання:

1. Контрольна робота 1: РН1.1, РН1.2, РН 2.1, РН3.1 – 30 балів/18 балів.
2. Контрольна робота 2: РН1.1, РН1.2, РН2.1, РН4.2 – 30 балів/18 балів.

Типова контрольна робота 1 складається з теоретичних та практичних завдань (з відкритими та закритими відповідями) за матеріалом розділу 1.

Матеріал, що виноситься на контрольну роботу 1:
див. запитання 1-30 для підготовки до оцінювання.

Типова контрольна робота 2 складається з теоретичних та практичних завдань (з відкритими та закритими відповідями) за матеріалом розділів 1 та 2.

Матеріал, що виноситься на контрольну роботу 2:
див. запитання 1-69 для підготовки до оцінювання.

- підсумкове оцінювання (у формі іспиту):

- 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: задачі (60%), теоретичне питання (40%, питання 1-69).

Студент допускається до іспиту, якщо в семестрі набрав не менше ніж 20 балів.

Питання до оцінювання

1. Поняття та властивості напівстатичних структур даних. Навести приклади.
2. Алгоритми роботи стека. Перерахуйте галузі застосування. Наведіть приклади реалізації.
3. Схема обробки черги і деку та їх складові частини. Реалізація черги та деку на базі масиву та динамічної структури.
4. Схема обробки стеку. Реалізація стеку на базі масиву та динамічної структури.
5. Бінарні дерева. Зображення і способи обходу. Наведіть приклади реалізації.
6. D- дерева. Зображення і галузі використання. Наведіть приклади реалізації.
7. Алгоритми, що визначає число елементів в бінарному дереві, які рівні заданому. Навести приклади.
8. Методи та алгоритми, що визначає число вузлів в бінарному дереві на заданому рівні. Навести приклади.
9. Методи та алгоритми пошуку елемента в бінарному дереві з заданою вагою. Навести приклади.
10. Методи та алгоритми для додавання та видалення елемента червоно-чорного дерева. Навести приклади.
11. Методи та алгоритми для обчислення висоти червоно-чорного дерева. Навести приклади.
12. Операції створення, друку та обходу дерева. Навести приклади.
13. Означення бінарних впорядкованих дерев, випадкові, оптимальні та збалансовані по висоті дерева. Навести приклади.
14. Поняття кодових дерев. Алгоритм Хаффмана. Навести приклади.
15. Бінарна піраміда як структура даних та її властивості.
16. Реалізація черги з пріоритетами на основі піраміди.
17. Випадкова побудова бінарних дерев пошуку.
18. Дерева відрізків.
19. Дерева Хаффмана і коди Хаффмана.
20. Елементарні структури даних (стеки, черги, зв'язані списки).
21. Способи представлення дерев з коренем.
22. Дерево пошуку як структура даних, операції над ним.
23. Червоно-чорні дерева, їх властивості, представлення і аналіз. Повороти. Відновлення червоно-чорних властивостей.
24. Піраміди злиття.
25. Біноміальні дерева, лема про їх властивості.
26. Біноміальна піраміда. Представлення біноміальних пірамід.
27. Основні операції над біноміальними пірамідами. Злиття біноміальних пірамід.
28. Піраміди Фібоначчі: означення і представлення. Функція потенціалу для пірамід Фібоначчі.
29. Основні операції над пірамідами Фібоначчі.
30. Оцінка максимальної степені вузла в піраміді Фібоначчі.
31. Який граф називається дводелевим?
32. Який гіперграф називається двоїстим до заданого?
33. Як змінюється матриця інцидентності графа чи орграфа при перенумерації його вершин? ребер (дуг)?
34. Як змінюється матриця суміжності вершин графа чи орграфа при перенумерації його вершин? ребер (дуг)?
35. Що таке паросполучення? максимальне за включенням? максимальне? зважене?
36. Що таке незалежна множина вершин? максимальна за включенням? максимальна? зважена?

37. Що таке реберне покриття? мінімальне за виключенням? мінімальне? зважене?
38. Що таке вершинне покриття? мінімальне за виключенням? мінімальне? зважене?
39. Що таке домінуюча множина ребер? мінімальна за виключенням? мінімальна? зважена?
40. Що таке домінуюча множина вершин? мінімальна за виключенням? мінімальна? зважена?
41. Що таке кліка? максимальна за включенням? максимальна? зважена?
42. Що таке розфарбовка вершин графа? правильна розфарбовка? мінімальна правильна розфарбовка? хроматричне число?
43. Що таке розфарбовка ребер графа? правильна розфарбовка? мінімальна правильна розфарбовка? хроматричний індекс?
44. Як формулюється теорема Візінга?
45. Що таке матроїд? носій матроїда? незалежна множина матроїда? залежна множина матроїда? база матроїда?
46. Що таке мінімальне остовне дерево? ліс?
47. Чому для побудови мінімального остовного дерева можна застосовувати жадібний алгоритм?
48. Які ви знаєте необхідні та достатні умови ейлеровості? напівейлеровості?
49. Які ви знаєте алгоритми побудови ейдерового циклу (шляху)?
50. Який граф називається гамільтоновим? напівгамільтоновим?
51. Які ви знаєте достатні умови гамільтоновості?
52. Які прості цикли називаються незалежними? скільки їх? як їх знайти?
53. Які коцикли називаються незалежними? скільки їх? як їх знайти?
54. Яким є бінарне відношення \rightarrow (досягається) з точки зору його класифікації?
55. Як будується матриця транзитивної досяжності?
56. Як знаходяться компоненти сильної зв'язності?
57. Як проводиться часткове упорядкування компонент сильної зв'язності?
58. Що таке ексцентриситет вершини графа?
59. Що таке радіус графа? діаметр графа?
60. Які вершини графа називаються центральними? периферійними?
61. Чим мережа відрізняється від орграфа? Як перетворити орграф на мережу?
62. Як формулюється задача про максимальний потік у мережі з обмеженими пропускними здатностями?
63. Яка задача є двоїстою до задачі про максимальний потік? Як вона формулюється?
64. Що таке ізоморфізм?
65. Що таке перестановка?
66. Що таке інваріант графа?
67. Що таке індекс Вінера?
68. Що таке індекс Рандича?
69. Що таке характеристичний поліном?

7.2 Організація оцінювання:

Терміни проведення форм оцінювання:

1. Контрольна робота 1: до останнього лекційного заняття.
2. Контрольна робота 2: до останнього лекційного заняття.

Перескладання контрольних робіт не передбачаються.

7.3 Шкала відповідності оцінок

Відмінно / Excellent	90-100
Добре / Good	75-89
Задовільно / Satisfactory	60-74

8. Структура навчальної дисципліни. Тематичний план лекцій

№ п/п	Назва теми	Кількість годин		
		Лекції	Лабораторні	Самостійна робота
Розділ 1. Базові структури даних.				
1.	Тема 1. Спеціалізовані структури даних: масиви, списки, стеки, черги. Представлення спеціалізованих структур за допомогою масивів.	1		5
2.	Тема 2. Алгоритми на деревах. Коди Хаффмана. Обчислення арифметичних виразів. Реалізація алгоритмів на деревах мовами програмування.	1		5
3.	Тема 3. Дерева пошуку. Бінарні дерева пошуку. d – дерева. Реалізація алгоритмів обходу дерев мовами програмування.	1		5
4.	Тема 4. Збалансовані дерева. Червоно-чорні дерева. AVL -дерева. В – дерева. Реалізація алгоритмів балансування дерев мовами програмування.		1	5
5.	Тема 5. Дерево Фенвіка. Дерево відрізків. Реалізація дерева Фенвіка мовами програмування			5
6.	Тема 6. Піраміди. Біноміальні піраміди, піраміди Фібоначчі. Тонкі та товсті піраміди. Лінійні лівосторонні піраміди. Реалізація пірамід мовами програмування.			5
	<i>Контрольна робота 1</i>			10
Всього по розділу 1				
Розділ 2. Алгоритми на графах.				
7.	Тема 7. Теорія графів. Основні властивості та поняття. Основні представлення графів: матриця суміжності, матриця інциденсій.	1		5
8.	Тема 8. Пакування, покриття, домінуючі множини, кліки. Двоїстість задачі про пакування та покриття.	1		6
9.	Тема 9. Правильне розфарбування графа. Реалізація алгоритмів розфарбування мовами програмування	1		6
10.	Тема 10. Матроїди та жадібні алгоритми. Мінімальне остовне дерево. Реалізація алгоритмів побудови мінімальних остовних дерев мовами програмування	1		6
11.	Тема 11. Задача Штейнера Реалізація задачі Штейнера мовами програмування		1	6

12.	Тема 12. Цикли та коцикли Побудова фундаментальної системи циклів та коциклів мовами програмування.			6
13.	Тема 13. Сильнозв'язані компоненти орграфу Визначення та впорядкування сильнозв'язаних компонент мовами програмування		1	6
14.	Тема 14. Мережеві задачі. Реалізація алгоритмів знаходження максимального потоку мовами програмування			6
15.	Тема 15. Інваріанти графів. Індокси Вінера, Рандича. Гіпотеза Улама про реконструкцію графів.			6
16.	Тема 16. Випадкові графи. Реалізація алгоритмів побудови випадкових графів мовами програмування			6
	<i>Контрольна робота 2</i>			10
Всього по розділу2				
ВСЬОГО		7	3	109

Загальний обсяг 120 год., в тому числі:

Лекцій – 7 год.

Лабораторні - 3 год.

Консультації – 1 год.

Самостійна робота - 109 год.

9. Рекомендовані джерела:

Основні:

1. Ахо А. Структуры данных и алгоритмы. / А. Ахо, Д. Холкрофт, Д. Ульман. – М.: Вильямс, 2000.
2. Кормен Т. Алгоритмы: построение и анализ / Т. Кормен, Ч. Лейзерсон, Р. Ривест, К. Штайн. – М. : Вильямс, 2013.
3. Кнут Д. Искусство программирования для ЭВМ. Тома 1-3. / Д. Кнут. – М. Вильямс, 2000.
4. Седжвик Р. Фундаментальные алгоритмы на С. Части 1-5. / Р. Седжвик – М.: Диасофт, 2003.
5. Стивенс Р. Алгоритмы. Теория и практическое применение./ Р. Стивенс. – М.: Из-во «Э», 2016.
6. Уоррен Г.С. Алгоритмические трюки для программистов. / Г.С. Уоррен. – М.:Вильямс, 2014.
7. Хопкрофт, Дж. Введение в теорию автоматов, языков и вычислений. / Дж. Хопкрофт, Р. Мотвани, Дж. Ульман. – М.:Вильямс, 2002.

Додаткові:

8. Кристофидес Н. Теория графов. Алгоритмический подход. /Н. Кристофидес. - М.: Мир, 1978.
9. Ловас Л. Прикладные задачи теории графов. / Л. Ловас, М. Пламмер. - М.: Мир, 1998.
10. Липский В. Комбинаторика для программистов. / В. Липский. - М.: Мир, 1988.
11. Новиков Ф.А. Дискретная математика для программистов. / Ф.А. Новиков. - СПб.: Питер, 2001.
12. Део Н. Комбинаторные алгоритмы. / Н. Део, Э. Рейнгольд, Ю. Нивергельт. - М.: Мир, 1980.

13. Коротєєва Т.О. Алгоритми та структури даних. / Т.О. Коротєєва. – Львів. В-во Львівської політехніки, 2014.
14. Ткачук В.М. Алгоритми і структура даних./ В.М. Ткачук. - ІваноФранківськ: Видавництво Прикарпатського національного університету імені Василя Стефаника, 2016.
15. Себеста Р.У. Основные концепции языков программирования. / Р.У. Себеста. - Диалектика-Вильямс, 2001.
16. Уоткинс Д. Программирование на платформе .Net / Д. Уоткинс, М. Хаммонд, Б. Эйбрамз. - М. "Вильямс", 2003.
17. Лейнингем И. Освой самостоятельно Python / И. Лейнингем –М.:Вильямс, 2001.
18. Пашко А.О. Комп'ютерний менеджмент мобільних систем / А.О. Пашко, О.В. Соломаха, Ю.В. Погорілий // Техніко-технологічні аспекти розвитку та випробування нової техніки і технологій для сільського господарства України. Збірник наукових праць УкрЦВТ. Вип. 5(19). 2002.С.106-111.

10. Додаткові джерела:

<https://drive.google.com/file/d/13RkPGApXEwDE9WULu3U1H-85oho91yDi/view?usp=sharing>