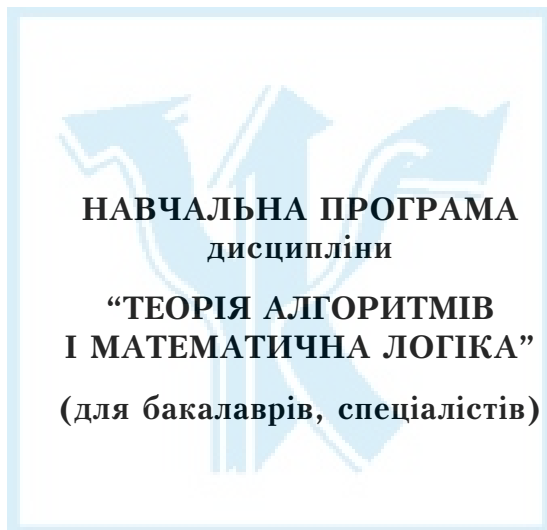


МІЖРЕГІОНАЛЬНА
АКАДЕМІЯ УПРАВЛІННЯ ПЕРСОНАЛОМ



НАВЧАЛЬНА ПРОГРАМА
дисципліни
**“ТЕОРІЯ АЛГОРИТМІВ
І МАТЕМАТИЧНА ЛОГІКА”**
(для бакалаврів, спеціалістів)

МАУП

Київ 2005

Підготовлено кандидатом фізико-математичних наук,
доцентом *І. К. Рисцовим*

Затверджено на засіданні кафедри інформатики та інформаційних
технологій (протокол № 9 від 27.10.04)

Схвалено Вченою радою Міжрегіональної Академії управління персоналом

Рисцов І. К. Навчальна програма дисципліни “Теорія алгоритмів і математична логіка” (для бакалаврів, спеціалістів). — К.: МАУП, 2005. — 16 с.

Навчальна програма містить пояснювальну записку, навчально-тематичний план, програмний матеріал до вивчення дисципліни “Теорія алгоритмів і математична логіка”, теми практичних занять, вказівки до виконання контрольної роботи, варіанти контрольних робіт, контрольні питання, а також список рекомендованої літератури.

© Міжрегіональна Академія
управління персоналом (МАУП), 2005

ПОЯСНЮВАЛЬНА ЗАПИСКА

Мета вивчення дисципліни “Теорія алгоритмів і математична логіка” — опанувати фундаментальним для інформатики поняттям алгоритму, сформувати практичні навички розробки алгоритмів розв’язання прикладних задач та їх програмування. Вивчення цієї дисципліни дасть змогу студентам зрозуміти та засвоїти основні принципи розробки алгоритмів і програм, а також стане підґрунтям для самостійної практичної роботи в галузі інформаційних систем.

Предметом дисципліни є комп’ютерна і обчислювальна математика. Крім того, у курсі вивчаються інформаційні структури даних, обчислювальні моделі й мови програмування високого рівня.

Навчальний курс передбачає знання, здобуті при вивченні шкільного курсу інформатики та дисципліни “Основи дискретної математики”.

На цьому курсі базуються курси програмування та чисельних методів в інформатиці.

НАВЧАЛЬНО-ТЕМАТИЧНИЙ ПЛАН *вивчення дисципліни*

“ТЕОРІЯ АЛГОРИТМІВ І МАТЕМАТИЧНА ЛОГІКА”

№ пор.	Назва теми
1	Алгоритми та їх властивості
2	Прикладна теорія алгоритмів
3	Теорія обчислень і математична логіка

ПРОГРАМНИЙ МАТЕРІАЛ *до вивчення дисципліни*

“ТЕОРІЯ АЛГОРИТМІВ І МАТЕМАТИЧНА ЛОГІКА”

Тема 1. Алгоритми та їх властивості

Неформальне поняття і визначення алгоритму. Алгоритм пошуку мінімуму в масиві як приклад алгоритму. Блок-схема алгоритму.

Основні властивості алгоритмів: функціональність, результативність, визначеність, елементарність. Алгоритмічні мови високого рівня. Оператор присвоювання. Умовний оператор. Оператор умовного та безумовного циклу. Виклик процедури.

Чисельні алгоритми. Алгоритм Евкліда пошуку найбільшого спільного дільника двох чисел. Доведення правильності та результативності алгоритму Евкліда. Алгоритм “помнож на три і додай одиницю” і проблемні (не результативні) алгоритми.

Алгоритм пошуку шляху в лабіринті (нитка Аріадни) як приклад алгоритму на графі. Операції зі стеком.

Література [1; 2; 4; 8]

Тема 2. Прикладна теорія алгоритмів

Основні етапи розробки алгоритму: постановка завдання і побудова моделі, розробка і реалізація алгоритму, доведення правильності та тестування алгоритму, аналіз складності алгоритму, підготовка документації.

Основні інформаційні структури даних: масиви, списки, черги, стеки. Зображення дерев, графів і множин.

Методи розробки алгоритмів: структурне програмування, рекурсія, обходи дерев, “поділяй і пануй”, балансування, динамічне програмування, програмування з відходом назад, метод “гілок і меж”, евристичні та наближені алгоритми.

Проходження повного циклу розробки на прикладі алгоритму пошуку кістякового дерева мінімальної ваги.

Алгоритми комп’ютерної математики. Алгоритми додавання і множення чисел. Аналіз та обчислення арифметичних і логічних виразів. Алгоритми на графах, пошук у глибину, пошук найкоротшого шляху. Сортування і пошук даних. Ігрові та комбінаторні алгоритми. Чисельні методи, обчислення дійсних констант і функцій. Ймовірнісні алгоритми. Алгоритми ідентифікації текстових рядків і скінченні автомати.

Література [1; 2; 4; 7; 9]

Тема 3. Теорія обчислень і математична логіка

Універсальні обчислювальні моделі: машини Тьюрінга і Поста, алгоритми Маркова, машини з вільним доступом до пам'яті, рекурсивні функції. Формальне визначення алгоритму, тезис Черча. Масові алгоритмічні проблеми, які неможливо розв'язати.

Елементи математичної логіки, обчислення висловлень, поняття логічного висновку, недетерміновані алгоритми. Теорема про повноту для обчислення висловлювань. Обчислення предикатів, формальна арифметика. Теорема Геделя про неповноту арифметики.

Література [3; 5; 6; 8]

ТЕМИ ПРАКТИЧНИХ ЗАНЯТЬ

1. Програмування алгоритму “помнож на три і додай одиницю”.
2. Програмування алгоритму пошуку виходу з лабіринту.
3. Математична індукція і перевірка правильності алгоритмів.
4. Зв'язані списки, вставлення і видалення елементів зі списку.
5. Робота зі стеком і чергами.
6. Алгоритмічне зображення множин, графів і дерев.
7. Структурне програмування “зверху до низу”. Теорема Якопіні.
8. Рекурсивні алгоритми, обчислення факторіалів та функції Акермана.
9. Рекурсивний обхід графа у глибину та ширину.
10. Порівняння рекурсії з ітерацією.
11. Алгоритми з відходом назад, тур коня і задача про вісім ферзів.
12. Метод часткових цілій і задача про джип.
13. Метод “поділяй і пануй”, надходження мінімумів і максимумів.
14. Динамічне програмування, алгоритм множення матриць.

15. Метод “гілок і меж”, задача про комівояжера.
16. Евристичні та наближені алгоритми.
17. Алгоритм пошуку кістякового дерева мінімальної ваги.
18. Булеві програми, комп'ютерна арифметика.
19. Алгоритми на графах; пошук найкоротшого шляху у графі.
20. Аналіз і обчислення арифметичних виразів; скобковий та польський запис.
21. Алгоритми сортування масивів; швидке сортування.
22. Ігрові алгоритми на прикладі ігор “нім” і “текс”.
23. Чисельні алгоритми; зображення дійсних чисел; плаваюча кома.
24. Ймовірнісні алгоритми; генератори випадкових чисел.
25. Скінченні та магазинні автомати; ідентифікація символічних рядків.
26. Машини з довільним доступом до пам'яті.
27. Машини Тьюрінга і алгоритми Маркова.
28. Рекурсивні функції, обчислення рекурсивних функцій.
29. Логічні формули і правила виведення, доведення висловлювань.
30. Предикати і квантори; доведення в теорії предикатів першого порядку.

ВКАЗІВКИ ДО ВИКОНАННЯ КОНТРОЛЬНОЇ РОБОТИ

Номер контрольної роботи студент визначає за останньою цифрою номера своєї залікової книжки, якщо ця цифра знаходиться між 1 і 5. Якщо ж цифра знаходиться між 6 і 9, то для визначення варіанта треба відняти від останньої цифри 5. Якщо номер залікової книжки закінчується на 0, студент виконує варіант 5. Студентам забороняється самостійно змінювати варіант контрольної роботи. У цьому разі робота може бути визнана недійсною.

На титульній сторінці контрольної роботи студент повинен написати своє прізвище, ім'я, індекс групи, номер залікової книжки, номер варіанта контрольної роботи.

Теоретичні питання студент повинен переписати і відповісти на них по суті в письмовому вигляді. Практичні завдання студент виконує на комп'ютері, а текст алгоритму і результати відоб-

ражає в роботі. Наприкінці контрольної роботи слід навести список використаної літератури, поставити дату та підпис.

Результати практичних завдань записують на дискету, додають її у конверт, який прикріплюють на останню сторінку контрольної роботи. На дискеті має бути наклейка, де вказуються такі самі дані, що й на титульній сторінці.

ВАРІАНТИ КОНТРОЛЬНИХ РОБІТ

Варіант 1

1. Наведіть визначення поняття “алгоритм”. Опишіть основні властивості алгоритмів.
2. Складіть блок-схему алгоритму знаходження суми числового масиву.
3. Опишіть алгоритм Евкліда знаходження найбільшого спільного дільника двох чисел. Запрограмуйте його однією з мов програмування (Паскаль, Си, VBA або Maple). Скільки кроків виконує цей алгоритм на парі чисел (120, 540) і якою буде відповідь? Файл з програмою, яку можна виконати, запишіть на дискету [4; 9].
4. Опишіть основні етапи побудови алгоритму [2].
5. Опишіть списки та алгоритми додавання і видалення елементів списку. Запрограмуйте ці алгоритми однією з мов програмування (Паскаль, Си або VBA). Файл з програмою, яку можна виконати, запишіть на дискету [1; 2; 4].
6. Опишіть алгоритм, який перевіряє правильність розташування лівих і правих дужок у виразі. Опишіть, як виконуватиметься цей алгоритм на ланцюжку (((())) ()) і яку відповідь він видасть? Запрограмуйте його однією з мов програмування (Паскаль, Си, VBA або Maple). Файл з програмою, яку можна виконати, запишіть на дискету [2].
7. Наведіть визначення поняття “рекурсивний алгоритм” і наведіть приклади. Обчисліть факторіал за допомогою рекурсивного алгоритму і запрограмуйте його однією з мов програмування (Паскаль, Си, VBA). Файл з програмою, яку можна виконати, запишіть на дискету [1; 2].

8. Опишіть алгоритм Крускала побудови кістякового дерева графа мінімальної ваги і запрограмуйте його однією з мов програмування (Паскаль, Си, VBA або Maple). Файл з програмою, яку можна виконати, запишіть на дискету [1; 2].
9. Опишіть, що називається машиною з довільним доступом до пам'яті. Чи можна на цих машинах обчислити будь-яку рекурсивну функцію? [1; 3].
10. Наведіть визначення поняття “тавтологія”. Доведіть, що логічна формула $(A \rightarrow B) \rightarrow ((B \rightarrow C) \rightarrow (A \rightarrow C))$ є тавтологією. Чи буде тавтологією формула $((A \rightarrow B) \rightarrow (B \rightarrow C)) \rightarrow (A \rightarrow C)$? [6; 7].

Варіант 2

1. Наведіть визначення поняття “алгоритмічна проблема”.
2. Складіть блок-схему алгоритму знаходження максимуму числового масиву.
3. Опишіть алгоритм “помнож на три і додай одиницю”. Запрограмуйте його однією з мов програмування (Паскаль, Си, VBA або Maple). Скільки кроків виконує цей алгоритм, якщо на його вхід надходить число 11? Файл з програмою, яку можна виконати, запишіть на дискету [8].
4. Опишіть, що називається перевіркою правильності алгоритму та його тестуванням. Доведіть, що алгоритм Евкліда знаходження найбільшого спільного дільника двох чисел правильний [2; 4; 9].
5. Опишіть стеки та основні операції з ними. Як у програмуванні використовуються стеки? [1; 2; 4].
6. Які структури даних використовуються для зображення графів у комп'ютері? Проілюструйте відповідь на прикладі графа, який має 5 вершин і ребра (1,2), (1,4), (2,3), (2, 4), (3,4) і (4,5). За допомогою алгоритму перетворіть список ребер цього графа на список суміжності вершин [1; 2].
7. Опишіть метод побудови алгоритмів “поділяй і пануй”. Проілюструйте його на прикладі алгоритму, який прискорює шкільний спосіб множення двох чисел. Запрогра-

- муйте цей алгоритм однією з мов програмування (Паскаль, Си або VBA). Файл з програмою, яку можна виконати, запишіть на дискету [1; 5].
8. Опишіть алгоритм Дейкстри знаходження шляху мінімальної ваги для кожної вершини графа і запрограмуйте його однією з мов програмування (Паскаль, Си, VBA або Maple). Файл з програмою, яку можна виконати, запишіть на дискету [1].
 9. Сформулюйте тезис Черча. Чи можна його довести? [3; 7; 9].
 10. Опишіть, що таке формула для обчислення предикатів першого порядку [7].

Варіант 3

1. Опишіть, що означає результативність алгоритму.
2. Складіть блок-схему алгоритму знаходження мінімуму числового масиву.
3. Опишіть алгоритм пошуку шляху між двома вершинами графа (нитка Аріадни). Запрограмуйте його однією з мов програмування (Паскаль, Си, VBA). Файл з програмою, яку можна виконати, запишіть на дискету [1; 9].
4. Для чого необхідно розробляти документацію на алгоритм? [2].
5. Опишіть черги та алгоритми додавання і видалення елементів з черги. Запрограмуйте ці алгоритми однією з мов програмування (Паскаль, Си або VBA). Файл з програмою, яку можна виконати, запишіть на дискету [2].
6. Опишіть метод пошуку розв'язку з відходом назад (backtracking). Проілюструйте його на прикладі алгоритму, який знаходить розв'язок задачі про вісім ферзів на шаховій дошці. Запрограмуйте цей алгоритм однією з мов програмування (Паскаль, Си або VBA). Файл з програмою, яку можна виконати, запишіть на дискету [2; 8].
7. Опишіть алгоритм, який перевіряє правильність розташування лівих і правих дужок трьох типів $()$, $[\]$, $\{ \}$ у виразі. Опишіть, як виконуватиметься цей алгоритм на ланцюжку символів $(() [] \{ })$ і яку відповідь він видає.

- Запрограмуйте його однією з мов програмування (Паскаль, Си або VBA). Файл з програмою, яку можна виконати, запишіть на дискету [2].
8. Опишіть алгоритм сортування злиттям числового масиву і запрограмуйте його однією з мов програмування (Паскаль, Си, VBA або Maple). Файл з програмою, яку можна виконати, запишіть на дискету [1; 8].
 9. Наведіть визначення примітивно рекурсивної функції. Доведіть, що додавання є примітивно рекурсивною функцією [3; 6; 7].
 10. Сформулюйте правило виведення *modus ponens* в обчисленні висловлювань і наведіть приклади його використання [6; 7].

Варіант 4

1. Опишіть, що означає детермінованість алгоритму.
2. Складіть блок-схему алгоритму знаходження середнього значення числового масиву.
3. Опишіть алгоритм Евкліда знаходження найбільшого спільного дільника двох чисел, не використовуючи операцію ділення. Запрограмуйте його однією з мов програмування (Паскаль, Си, VBA або Maple). Скільки кроків виконує цей алгоритм на парі чисел (110, 676) і якою буде відповідь? Файл з програмою, яку можна виконати, запишіть на дискету [9].
4. Для чого необхідно будувати модель алгоритмічної задачі? [2].
5. Наведіть визначення поняття “дерево”. Як зображується дерево в комп’ютері? Проілюструйте відповідь на прикладі дерева, яке має 5 вершин і ребра (1,2), (1,4), (2,3) і (4,5). Зобразьте це дерево масивом [1; 2].
6. Опишіть метод динамічного програмування. Проілюструйте його на прикладі алгоритму, який знаходить розв’язок задачі про оптимальну послідовність множення матриць. Запрограмуйте цей алгоритм однією з мов програмування (Паскаль, Си або VBA). Файл з програмою, яку можна виконати, запишіть на дискету [1].

7. Опишіть хвильовий алгоритм пошуку найкоротшого шляху між двома вершинами графа. Опишіть, як виконуватиметься цей алгоритм на графі $(1,2)$, $(1,4)$, $(2,3)$, $(2,4)$, $(3,4)$ і $(4,5)$ і яку відповідь він видасть, якщо треба знайти шлях між вершинами 1 і 5. Запрограмуйте його однією з мов програмування (Паскаль, Си або VBA). Файл з програмою, яку можна виконати, запишіть на дискету [8].
8. Запрограмуйте однією з мов програмування (Паскаль, Си, VBA або Maple) гру “Життя”, яка моделює клітинні автомати і процеси розмноження живих істот. Файл з програмою, яку можна виконати, запишіть на дискету [10].
9. Наведіть визначення рекурсивної множини натуральних чисел. Доведіть, що множина простих чисел є рекурсивною [3; 6; 7].
10. Наведіть одну із систем аксіом для обчислення висловлювань. Наведіть приклади виведення висловлювань з цієї системи [6; 7].

Варіант 5

1. Опишіть, що означає функціональність алгоритму і принцип “чорної шухляди” [2].
2. Складіть блок-схему алгоритму знаходження добутку числового масиву.
3. Опишіть алгоритм пошуку шляху між двома вершинами графа (нитка Аріадни), якщо граф заданий списком суміжності своїх вершин. Скільки часу (кроків) працюватиме цей алгоритм залежно від числа вершин і ребер графу? [1; 9].
4. Що передбачає етап реалізації алгоритму? [2].
5. Опишіть рекурсивний алгоритм нумерації вершин бінарного дерева згідно з внутрішнім порядком його вершин. Наведіть нерекурсивний аналог цього алгоритму зі стеком. Запрограмуйте цей алгоритм однією з мов програмування (Паскаль, Си або VBA). Файл з програмою, яку можна виконати, запишіть на дискету. Проілюструйте його роботу на прикладі якогось бінарного дерева з 7 вершинами [1].

6. Опишіть евристичний метод побудови алгоритмів. Проілюструйте його на прикладі задачі про комівояжера (пошуку оптимального циклу в навантаженому повному графі). Запрограмуйте цей алгоритм однією з мов програмування (Паскаль, Си, VBA або Maple). Файл з програмою, яку можна виконати, запишіть на дискету [2].
7. Опишіть алгоритм перетворення арифметичного виразу, який містить операції $+$, $-$, $*$ і $/$ на польський бездужковий запис. Запрограмуйте його однією з мов програмування (Паскаль, Си, VBA або Maple). Файл з програмою, яку можна виконати, запишіть на дискету [8].
8. Опишіть, що означають ймовірнісні алгоритми і генератори псевдовипадкових чисел. Опишіть і запрограмуйте один із цих генераторів. Файл з програмою, яку можна виконати, запишіть на дискету [2; 8].
9. Наведіть визначення рекурсивно переліченої множини натуральних чисел. Наведіть приклади рекурсивно перелічених але не рекурсивних множин [3; 6; 7].
10. Наведіть одну з систем аксіом для формальної арифметики [6; 7].

КОНТРОЛЬНІ ПИТАННЯ

1. Що називається алгоритмом? Основні властивості алгоритмів.
2. З яких елементів складається блок-схема алгоритму?
3. Основні конструкції алгоритмічних мов високого рівня.
4. Які форми має умовний оператор?
5. Які форми має оператор циклу?
6. Алгоритм Евкліда знаходження найбільшого спільного дільника двох чисел.
7. Довести правильність алгоритму Евкліда.
8. Алгоритм “помнож на три і додай одиницю”.
9. Алгоритм пошуку виходу з лабіринту. Що є ниткою Аріадни в цьому алгоритмі?
11. Основні етапи побудови алгоритму.
12. З якою метою будують алгоритмічну модель задачі?

13. Як реалізувати алгоритм?
14. Що означає перевірка правильності алгоритму?
15. Що означає доведення правильності алгоритму?
16. Що означає тестування алгоритму?
17. Що передбачає аналіз алгоритму на складність?
18. Що таке документування алгоритму?
19. Що таке список і чим він відрізняється від масиву?
20. Основні операції, які можна виконувати зі списками.
21. Що таке стек і які операції можна з ним виконувати?
22. Які структури даних використовуються для зображення множин?
23. Як зображувати бінарні дерева і як їх обходити?
24. Які структури даних використовуються для зображення графів?
25. Як обходити граф у глибину і ширину?
26. Що таке рекурсивний алгоритм?
27. Порівняйте рекурсивні та ітераційні алгоритми?
28. Що означає принцип “поділяй і пануй” при побудові алгоритмів?
29. Що означає принцип балансування в теорії алгоритмів?
30. Алгоритм динамічного програмування.
31. Що означає алгоритм спрямованого перебору?
32. Що таке метод гілок і границь?
33. Що таке евристичний алгоритм?
34. Основні алгоритми сортування даних.
35. Що називається польським записом алгебраїчного виразу і які існують види цих записів? Наведіть приклади.
36. Алгоритм обчислення алгебраїчного виразу в польському запису.
37. Основні алгоритми на графах.
38. На прикладі алгоритму побудови кістякового дерева мінімальної вартості покажіть етапи розробки алгоритму.
39. Як визначити зв'язність графа?
40. Як розв'язати задачу пошуку найкоротших маршрутів у графі?
41. Ітераційний алгоритм пошуку кореня рівняння.

42. Що називається ймовірнісним алгоритмом? Наведіть приклади.
43. Що називається методом Монте-Карло розв'язання математичних задач?
44. Генератори випадкових чисел. Наведіть приклади.
45. Що називається машиною з довільним доступом до пам'яті?
46. Що називається машиною Тьюрінга?
47. Що називається алгоритмом Маркова?
48. Що називається рекурсивною функцією?
49. Що затверджується в тезису Черча і чи можна довести цей тезис?
50. Що називається алгоритмічною проблемою, яку не можна розв'язати.
51. Що називається рекурсивною і рекурсивно переліченою множиною чисел?
52. Наведіть приклади рекурсивно перелічених, але не рекурсивних множин.
53. Що називається тавтологією?
54. Система аксіом обчислення висловлювань.
55. Що називається правилом виведення *modus ponens*?
56. Що називається формальним доведенням у математичній логіці?
57. Теорема про повноту обчислення висловлювань.
58. Що називається формулою для обчислення предикатів першого порядку?
59. Теорема про повноту обчислення предикатів.
60. Теорема Геделя про неповноту арифметики.

СПИСОК РЕКОМЕНДОВАНОЇ ЛІТЕРАТУРИ

1. Ахо А., Хопкрофт Дж., Ульман Дж. Построение и анализ вычислительных алгоритмов. — М.: Мир, 1979. — 540 с.
2. Гудман С., Хидетниеми С. Введение в разработку и анализ алгоритмов. — М.: Мир, 1981. — 366 с.
3. Катленд Н. Вычислимость. Введение в теорию рекурсивных функций. — М.: Мир, 1983. — 256 с.

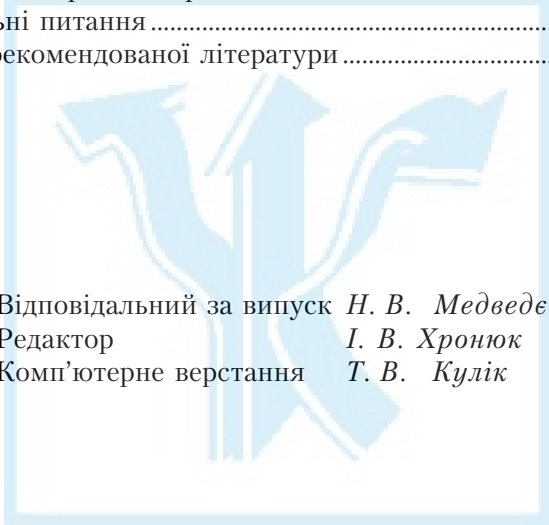
4. *Кнут Д.* Искусство программирования для ЭВМ. Основные алгоритмы. — М.: Мир, 1976. — Т. 1. — 736 с.
5. *Кнут Д.* Искусство программирования для ЭВМ. Получисленные алгоритмы. — М.: Мир, 1977. — Т. 2. — 724 с.
6. *Лавров И. А., Максимова Л. Л.* Задачи по теории множеств, математической логике и теории алгоритмов. — М.: Наука, 1975. — 240 с.
7. *Мендельсон Э.* Введение в математическую логику. — М.: Наука, 1971. — 320 с.
8. *Нивергельт Ю., Фаррар Дж., Рейнгольд Э.* Машинный подход к решению математических задач. — М.: Мир, 1977. — 352 с.
9. *Трахтенброт Б. А.* Алгоритмы и вычислительные автоматы. — М.: Сов. радио, 1974. — 200 с.
10. *Уэзерелл Ч.* Этюды для программистов. — М.: Мир, 1982. — 284 с.



МАУП

ЗМІСТ

Пояснювальна записка	3
Навчально-тематичний план вивчення дисципліни “Теорія алгоритмів і математична логіка”	3
Програмний матеріал до вивчення дисципліни “Теорія алгоритмів і математична логіка”	3
Теми практичних занять	5
Вказівки до виконання контрольної роботи	6
Варіанти контрольних робіт	7
Контрольні питання	12
Список рекомендованої літератури	14



Відповідальний за випуск *Н. В. Медведєва*
Редактор *І. В. Хролюк*
Комп'ютерне верстання *Т. В. Кулік*

Зам. № ВКЦ-2021

Міжрегіональна Академія управління персоналом (МАУП)
03039 Київ-39, вул. Фрометівська, 2, МАУП