1. Дано N прямокутників зі сторонами, що паралельні осям координат. Потрібно визначити, на скільки частин ці прямокутники розбивають площину.

2. а) Створити АВЛ-дерево.

б) Створити червоно-чорне дерево.

Реалізувати операції вставки, видалення елементів у створених деревах.

3. Розташовану вертикально прямокутну паперову стрічку з закріпленим нижнім кінцем стали складувати наступним чином.

На першому етапі її зігнули навпіл так, що верхня половина лягла на нижню або попереду (П-згинання) або позаду (З-згинання); на наступних n-1 кроках виконували аналогічну дію з отриманою на попередньому кроці зігнутою стрічкою як з єдиним цілим.

Потім стрічку розгорнули, привівши її у початковий стан. На ній залишилися згини - ребра від перегинів. Деякі з ребер виявилися спрямованими опуклістю до нас (К-ребра), а деякі - від нас (О-ребра). Ребра пронумерували зверху вниз числами від 1 до 2n-1.

а) Скласти програму, яка запитує послідовність символів з прописних літер "П" і "З" (вони визначають типи згинань) і видає послідовність типів ребер (у порядку зростання їх номерів), що вийшов після заданої послідовності згинань.

б) Скласти програму, яка запитує послідовність символів з прописних літер "О" і "К", де знаходження на i-тому місці символу "О" або "К" визначає тип ребра на розправленій стрічці, і видає рядок з прописних "П" і "З", що визначають послідовність типів згинань, за допомогою яких отримана стрічка з вихідною послідовністю ребер. Якщо такого рядка не існує, повідомити про це.

4. Методом Монте-Карло обчислити 2-кратний інтеграл від функції

f(x, y) = x\*x + y\*y. Область інтегрування: 0.5 <= x <= 1, 0 <= y <= 2x-1.

5. Створити однозв'язний лінійний список (у кожного вузла одне інформаційне поле типу Integer) з можливістю додавати і видаляти вузли.

Після завершення редагування списку запросити число N і розбити список на два підсписки (не змінюючи розташування елементів в пам'яті!) у такий спосіб: якщо значення інформаційного поля вузла більше N, то включити його в перший список, інакше - у другий список. Після завершення роботи усі списки видалити.

6. По колу розміщено N монет гербами вгору і M монет гербами вниз. Обходячи коло по ходу годинникової стрілки, перевертається кожна S-та монета. У перший раз рахунок починається з герба. У якому порядку потрібно розставити монети, щоб після K ходів стало L монет, що лежать гербами вгору.

7. Реалізувати роботу динамічної структури ЧЕРГА з можливістю додавання, видалення і сортування елементів. У роботі використовувати два методи сортування: швидке і вставками; порівняти ефективність цих методів.

8. Інтерполяція. Функція f(x) задана таблицею своїх значень:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| х | 0.5 | 1.5 | 2.0 | 3.5 | 6.0 | 9.0 |
| f(x) | 2.5 | 6 | -3.5 | -7 | -10 | 2 |

Побудувати кубічний сплайн і оцінити значення функції у точці x0 = 2.5

9. Реалізувати алгоритм перевірки графа на зведення (рос. сводимость).

10. Знайти ейлерів цикл у графі, що не містить вершин непарного степеня. Граф задається списком інцидентності у текстовому файлі. Кожен рядок містить пари: (ребро, вершина). У файлі не повинно бути порожніх рядків. При пошуку циклу використовується пошук у глибину.

11. Обходом по контуру заданий опуклий n-кутник (n => 3). Розбити його на трикутники (n-3)-ма діагоналями, що перетинаються лише у кінцях, таким чином щоб

а) сума їх довжин була мінімальною;

б) максимальна з діагоналей мала найменшу довжину.

12. Видалити з матриці [А] рядок і стовпець, що містять найбільший елемент матриці. Матриця [А] є розрядженою і зберігається у вигляді мультисписку.

13. Дана прямокутна таблиця, що складається з m рядків і n стовпців. На перетині i-го рядка і j-го стовпця записано ціле число aij. Потрібно знайти такі чотири різні комірки таблиці, щоб вони були вершинами прямокутника зі сторонами, паралельними сторонам таблиці, а сума чисел, записаних в цих комірках, була максимальна.

14. Написати програму, яка здійснює пірамідальне сортування масиву з використанням рекурсії.

15. Використовуючи метод пошуку в ширину, знайти найкоротший шлях від початкової до будь-якої довільної вершини зв'язкового неорієнтованого графа, заданого списками інцидентності (ваги всіх ребер прийміть рівними одиниці).

16. Написати програму, яка здійснює пошук в кореневому дереві піддерев, ізоморфних заданому дереву (зразку). Використати алгоритм повного обходу дерева з бектрекінгом (backtracking).

17. Знайти бікомпоненти графа (бікомпонента (Strongly connected component) - максимальний по включенню вершин сильно зв'язний підграф орграфа. Інша назва - компонента сильної зв'язності), заданого списками суміжності.

18. Використовуючи алгоритм пошуку в глибину, написати програму, яка знаходить множину фундаментальних циклів зв'язкового неорієнтованого графа, заданого таблицею інцидентності.

19. Для даного графа знайти мінімальний кістяк за допомогою:

а) алгоритму Прима;

б) алгоритму Крускала.

20. Кожне число являє собою масив байт.

Перший байт масиву містить код знака числа: 0 - "+", 225 - "-", а решта байтів- значущі двійкові цифри числа. У програмі розробити процедури алгебраїчних операцій з двома цілими числами, кожне з яких представлено масивом байтів.

Реалізувати операції: додавання, віднімання, множення, ділення на ціле. При реалізації операцій врахувати можливість переповнення.

21. У графі, заданому списками інцидентності, знайти всі гамаки.

22. Вирішити завдання комівояжера, використовуючи генетичний алгоритм.

23. Написати програму, яка в діалоговому режимі:

а) створює таблицю,

б) додає, видаляє стовпці та рядки,

в) читає з файлу і записує у файл,

г) редагує стовпці. У стовпців можуть бути типи: integer, short, char, string, double, bool. Типи можна змінювати і перетворювати (якщо можливо, наприклад short в integer).

24. Реалізувати алгоритм стиснення Хаффмана.

25. З даного рядка символів видалити найменшу кількість символів, так, щоб вийшов паліндром (рядок, який однаково читається як справа наліво, так і зліва направо).

26. (Завдання на графи) Нехай заданий деякий словник слів. Назвемо сходинкою редагування таке перетворення слова *r* у слово *q*, що слова *r* і *q* належать словнику і слово *r* може бути перетворено у слово *q* шляхом додавання, видалення або зміни однієї літери. Перетворення dig в dog і dog в do є сходинками редагування. Драбинка сходинок редагування - це відсортована в лексикографічному порядку по слідовність слів w1, w2, ..., wn така, що перетворення wk в wk+1 є сходинкою редагування для будь-якого *k* від 1 до n-1. Для заданого словника знайти найбільшу драбинку сходинок редагування. При цьому на вхід програми подаються дані, що складаються з набору слів на нижньому регістрі в лексикографічному порядку по одному слову на рядок. Довжина кожного слова не перевищує 16 букв, і в словнику не більше 25 000 слів.

27. Побудувати М- і Т-нумерацію для заданого графа.

28. Реалізувати паралельний алгоритм сортування вичерпуванням і оцінити його часову складність.

29. Написати програму, яка обчислює (як ціле число) значення виразу (без змінних), що записаний у звичайній формі у текстовий файл expression.txt. Перед обчисленням вираз необхідно перевести у постфіксну форму. Постфіксною формою виразу a^b називається запис, у якому знак операції ^ розміщений за операндами: ab^. Наприклад, постфіксною формою виразу a+b-c буде вираз ab+c-, а виразу a+b\*c - вираз bc\*a+.

30. Реалізувати чергу з використанням алгоритму, запропонованого Меджедом М. Майклом і Майклом Л. Скоттом.

31. Послідовність сум {sn}, де sn = 1 - x2/2! + ... + (-1)nх2n/(2n)!, за умови |х| < π/4 "достатня швидко" сходиться до cos(x).

а) Запрограмувати обчислення cos(x) при х[-π/4; π/4] з точністю до ε, тобто за потрібне число приймає перше sn таке, що |sn - sn-1| < ε.

б) Запрограмувати обчислення cos(x) при довільному х, застосувавши тригонометричні формули зведення.

32. Дано два рядки *a* і *b.* Вивести рядок *х* максимальної довжини, що складається з букв таких, що існує перестановка *х*, що є підрядком перестановки *a* та одночасно є підрядком перестановки *b*.

33. Реалізувати алгоритм рішення загального лінійного діафантового рівняння.

34. Реалізувати основні операції роботи з B-деревом (додавання ключа, пошук ключа, видалення ключа).

35. (Завдання на графи). Відсортована у порядку зростання послідовність різних величин може бути сформована, якщо над цими величинами встановлено відношення порядку (<). Наприклад, відсортована за зростанням послідовність А, В, С, D означає, що А<В, В<С і С<D. Вам дається множина відношень виду А<В, і потрібно встановити, чи можна сформувати зростаючу послідовність.

36. На столі лежить купка з N сірників. Гравець 1 із Гравцем 2 по черзі беруть сірники з купки. Першим бере Гравець 1 і йому дозволяється взяти від 1 до К сірників. Після цього, починаючи з Гравця 2, гравці можуть брати будь-яку кількість сірників, що не більше ніж на 1 перевищує ту кількість, яку взяв гравець перед ним (можна взяти менше або стільки ж, але обов'язково хоча б один). Наприклад, якщо N = 10, К = 5, то на першому ході Гравець 1 може взяти 1, 2, 3, 4 або 5 сірників; якщо Гравець 1 візьме 3, то на наступному ході Гравець 2 може взяти 1, 2, 3 або 4; і якщо Гравець 2 візьме 1, то Гравець 1 потім може взяти 1 або 2, і т.д. Програє той, хто візьме останній сірник. Визначити, скільки сірників повинен взяти на першому ході Гравець 1, щоб виграти при будь-якій грі Гравця 2.

37. Дана L-система (аксіома, правила, інтерпретація).На базі цієї L побудувати фрактал відповідного порядку.

38. а) Знайти найкоротшу відстань між двома вершинами в графі.

б) Знайти всі можливі шляхи між цими двома вершинами, що не пеpетинаються: а) по pебpам; б) по вершинам.

39. Є N людей і прямокутна таблиця А[1:N, 1:N]; елемент A[i, j] дорівнює 1, якщо людина *i* знайома з людиною *j*, А[i, j] = А [j, i]. Чи можна розбити людей на дві групи так, щоб у кожній групі були тільки незнайомі люди.

40. Написати інтерпретатор мови А

ОПИС МОВИ А

Програма на мові А - це послідовність операторів, кожен з яких записується в окремому рядку. Порожні рядки допускаються. Мова включає оператор присвоювання, оператор переходу, умовний оператор, оператор введення, оператор виведення і оператор зупинки.

=============

Оператор присвоювання має вигляд: ІМ'Я:=ВИРАЗ

де ІМ'Я - ім'я змінної, що складається з однієї латинської букви, всі змінні мови мають цілий тип і приймають значення від -1000 до 1000, змінні явно не описуються і за замовчуванням мають нульові значення.

ВИРАЗ - це цілочисловий вираз, що складається з операндів і знаків операцій. Операндами можуть бути константи, імена змінних і покажчики вбудованих функцій. Константами мови є звичайні цілі десяткові константи від -1000 до 1000.

Мова містить одну вбудовану функцію abs, що повертає абсолютну величину числового аргументу.

Мова включає 4 операції: "+" - додавання, "-" - віднімання, "\*" - множення і ":" - цілочислове ділення.

Пріоритети операцій "\*" і ":" вищі, ніж пріоритети операцій "+" і "-"; операції одного пріоритету виконуються у послідовності зліва направо, дужки у виразах не допускаються.

=============

Оператор переходу: go МІТКА

де МІТКА - мітка будь-якого оператора програми, що представляє собою ціле число від 0 до 1000. Назва може бути вказана на початку будь-якого оператора і відділяється від оператора символом "#".

=============

Умовний оператор:

if (ВІДНОШЕННЯ) ОПЕРАТОР

де ВІДНОШЕННЯ - це конструкція, що складається з двох виразів, розділених знаком відношення; знаки відносини - це "=", "! =", ">", "> =", "<", "< =".

ОПЕРАТОР - це будь-який оператор крім умовного.

=============

Оператор введення: get(ІМ'Я)

де ІМ'Я - ім'я змінної, що вводиться значення набирається на клавіатурі у вигляді правильної константи мови А. Кожен новий оператор get починає введення з нового рядка.

=============

Оператор виводу: put(ВИРАЗ/ЛІТЕРАЛ)

де ВИРАЗ - правильний вираз мови А, ЛІТЕРАЛ - конструкція $послідовність символів. При виведенні виразу на екран виводиться константа, що дорівнює його значенню, при виведенні літералу на екран виводяться символи, що його складають, без першого символу "$". Кожен оператор put виводить з нового рядка.

=============

Оператор зупинки: finish

зупиняє роботу програми.