

Список задач

1. Дано N прямоугольников со сторонами, параллельными осям координат. Требуется определить, на какое количество частей эти прямоугольники разбивают плоскость.
2. Реализовать АВЛ –дерево.
Реализовать красно-черное дерево.
Необходимо реализовать операции вставки, удаления элементов из соответствующих деревьев.
3. Расположенную вертикально прямоугольную бумажную ленточку с закрепленным нижним концом стали складывать следующим образом:
 - на первом шаге ее согнули пополам так, что верхняя половина легла на нижнюю либо спереди (П - сгибание) либо сзади (З сгибание),
 - на последующих $n-1$ шагах выполняли аналогичное действие с получающейся на предыдущем шаге согнутой ленточкой, как с единым целым.Затем ленточку развернули, приведя ее в исходное состояние. На ней остались сгибы - ребра от перегибов, причем некоторые из ребер оказались направленными выпуклостью к нам (К - ребра), а некоторые - от нас (О -ребра). Ребра пронумеровали сверху вниз числами от 1 до 2^{n-1} .
- А. Составить программу, запрашивающую строку символов из прописных букв "П" и "З", определяющую последовательность типов сгибаний, - номер ребра, и сообщаемую тип этого ребра, получившийся после заданной последовательности сгибаний.
- Б. Составить программу, запрашивающую строку символов из прописных букв "О" и "К", где нахождение на i -том месте символа "О" или "К" определяет тип ребра на расправленной полоске, и выдающую строку из прописных "П" и "З", определяющих последовательность типов сгибаний, посредством которых получена ленточка с исходной последовательностью ребер. Если такой строки не существует, сообщить об этом.
4. Вычислить методом Монте-Карло 2-кратный интеграл $f(x,y) = x*x + y*y$
Область интегрирования: $0.5 \leq x \leq 1, 0 \leq y \leq 2x-1$
5. Создать односвязный линейный список (у каждого узла 1 информ. поле типа Integer), с возможностью добавлять и удалять узлы.
После завершения редактирования списка запросить число N и разбить список на два других, не меняя расположения элементов в памяти, следующим образом: если значение информ. поля узла $> N$, включить его в 1 список, иначе включить его во 2 список.
После завершения работы все списки разрушить.
6. По кругу расположено N монет гербами вверх и M монет гербами вниз. Обходя круг по ходу часовой стрелки, переворачивается каждая S -тая монета. В первый раз счет начинается с герба. В каком порядке надо расставить монеты, чтобы после K ходов стало L монет, лежащих гербами вверх.

7. Реализовать работу динамической структуры: ОЧЕРЕДЬ
Добавление элемента·Удаление·Сортировка: В работе использовать два метода сортировки: быструю и вставками. Сравнить эффективность.

8. Интерполирование.

Функция $f(x)$ задана таблицей своих значений. Построить кубический сплайн и оценить значение функции в указанной точке x_0 .

x -0.5 1.5 2.0 3.5 6.0 9.0

$f(x)$ 2.5 6 -3.5 -7 -10 2

$x_0=2.5$

9. Реализовать алгоритм проверки графа на сводимость.
10. Найти эйлеров цикл в графе, не содержащем вершин нечетной степени и заданном списками инцидентности.
- Граф задается списком инцидентности в текстовом файле.
- Каждая строка содержит пару: ребро вершина
- В файле не должно быть пустых строк.
- При поиске цикла используется поиск в глубину.
11. Дан выпуклый n -угольник, $n \geq 3$, своим обходом по контуру. Разбить его на треугольники $(n-3)$ -мя диагоналями, непересекающимися кроме как по концам, таким образом чтобы
а) Сумма их длин была минимальной;
б) Максимальная из диагоналей имела наименьшую длину.
12. Удалить из матрицы $[A]$ строку и столбец, содержащие наибольший элемент матрицы. Матрица $[A]$ является разреженной и хранится в виде мультисписков.
13. Дана прямоугольная таблица, состоящая из m строк и n столбцов. На пересечении i -й строки и j -го столбца записано целое число a_{ij} . Требуется найти такие четыре различные ячейки таблицы, чтобы их центры были вершинами прямоугольника со сторонами, параллельными сторонам таблицы, а сумма чисел, записанных в этих ячейках, была максимальна.
14. Написать программу, которая осуществляет пирамидальную сортировку массива с использованием рекурсии.
15. Используя метод поиска в ширину, найдите кратчайший путь от начальной до любой произвольной вершины связного неориентированного графа, заданного списками инцидентности (веса всех ребер примите равными единице).

16. Написать программу, которая осуществляет поиск в корневом дереве поддеревьев, изоморфных заданному искомому дереву.
Использовать алгоритм полного обхода дерева с бэктрекингом (backtracking).
17. Найти бикомпоненты графа заданного списками смежности.
18. Используя алгоритм поиска в глубину, написать программу, которая находит множество фундаментальных циклов связного неориентированного графа, заданного таблицей инцидентности.
19. Для данного графа найти минимальное остовное дерево. Использовать алгоритм Прима и алгоритм Крускала.
20. Каждое число представляет собой массив байт.
Первый байт массива содержит код знака числа: 0 - "+", 225 - "-", а остальные байты - значащие двоичные цифры числа. В программе разработать процедуры алгебраических операций с двумя целыми числами, каждое из которых представлено массивом байтов.
Операции:
1. Сложение
2. Вычитание
3. Умножение
4. Деление на цело
При выполнении операций учитывается возможность переполнения.
21. В графе, заданном списками инцидентности, найти все гаммаки.
22. Решить задачу коммивояжера используя генетический алгоритм.
23. Написать программу, которая в диалоговом режиме создаёт таблицу, которая, в свою очередь (тоже в диалоговом режиме) умеет:
1 Создавать, добавлять, удалять столбцы, строки и ячейки
2 Читаться и записываться в файл
3 Редактировать столбцы
У столбцов могут быть типы: integer, short, char, string, double, bool.
Типы можно менять и преобразовывать при возможности, т.е. те, которые возможно. Например, short в integer.
24. Используя алгоритм Хаффмана, построить H-дерево по исходному тексту содержащемуся в файле fl.txt.
25. Из данной строки удалите наименьшее количество символов, так, чтобы получился палиндром (строка, одинаково читающаяся как справа налево, так и слева направо).
26. Назовем ступенькой редактирования такое преобразование слова x в слово y, что слова x и y принадлежат словарю и слово x может быть преобразовано в слово y путем добавления, удаления или изменения одной буквы. Преобразования dig в dog и dog в do являются ступеньками редактирования. Лесенка ступенек редактирования - это отсортированная в лексикографическом порядке

последовательность слов w_1, w_2, \dots, w_n такая, что преобразование w_i в w_{i+1} является ступенькой преобразования для любого i от 1 до $n-1$. Для заданного словаря найти самую большую лесенку ступенек редактирования.

На вход подаются данные, которые состоят из словаря: набора слов в нижнем регистре в лексикографическом порядке по одному слову на строку. Длина каждого слова не превышает 16 букв, и в словаре не более 25 000 слов (Задача на графы).

27. Построить М - и Т- нумерацию для заданного графа.
28. Реализовать алгоритм сортировки вычерпыванием и оценить его временную сложность
29. Написать программу, которая вычисляет как целое число значение выражения (без переменных), записанное в обычной форме в текстовый файл expression.txt. Перед вычислением выражение необходимо перевести в постфиксную форму. Постфиксной формой записи выражения a^b называется запись, в которой знак операции размещен за операндами: ab^{\wedge} .
Например:
 $a+b-c$ это $ab+c-$
 $a+b*c$ это $bc*a+$
30. Используя генетический алгоритм найти минимум функции одного переменного $f(x)$.
31. Последовательность сумм $\{s_n\}$ где $s_n = 1 - x^2/2! + \dots + (-1)^n x^{2n}/(2n)!$, при условии $|x| < \pi/4$ "достаточно быстро" сходится к $\cos(x)$. Запрограммировать вычисление $\cos(x)$ при $x \in [-\pi/4; \pi/4]$ с точностью ϵ , т.е. за нужное число принимается первое s_n такое, что $|s_n - s_{n-1}| < \epsilon$. Запрограммировать вычисление $\cos(x)$ при произвольном x , применив тригонометрические формулы сведения
32. Даны две строки a и b , вывести строку x максимальной длины, состоящую из букв, таких, что существует перестановка x , являющаяся подстрокой перестановки a и одновременно являющаяся подстрокой перестановки b .
33. Реализовать алгоритм решения общего линейного диофантового уравнения.
34. Реализовать основные операции работы с В-деревом (добавление ключа, поиск ключа, удаление ключа)
35. Отсортированная в порядке возрастания последовательность различных величин может быть сформирована, если над этими величинами установлено отношение порядка ($<$). Например, отсортированная по возрастанию последовательность A, B, C, D означает, что $A < B, B < C$ и $C < D$. Вам дается множество отношений вида $A < B$, и требуется установить, можно ли сформировать возрастающую последовательность. (Задача на графы)
36. На стол кладется кучка из N спичек, затем Игрок 1 с Игроком 2 по очереди берут спички из кучки. Первым берет Игрок 1, ему разрешается взять от 1 до K спичек. Затем игрок может взять любое количество спичек, не более чем на 1

превышающее то количество, которое взял игрок перед ним (можно взять меньше или столько же, но обязательно хотя бы одну). Например, если $N = 10$, $K = 5$, то на первом ходу Игрок 1 может взять 1, 2, 3, 4 или 5 спичек; если Игрок 1 возьмет 3, то на следующем ходу Игрок 2 может взять 1, 2, 3 или 4; и если Игрок 2 возьмет 1, то Игрок 1 затем может взять 1 или 2, и т. д. Проигрывает тот, кто возьмет последнюю спичку. Рассчитать, какое количество спичек должен взять на первом ходу Игрок 1, чтобы выиграть при любой игре Игрока 2.

37. Дана L-система(аксиома, правила, интерпретация). По этой L построить фрактал соответствующего порядка.
38. Найти кратчайшее расстояние между двумя вершинами в графе. Найти все возможные пути между этими двумя вершинами в графе не пересекающиеся по
- а) ребрам
 - б) вершинам
39. Имеется N человек и прямоугольная таблица $A[1:N, 1:N]$; элемент $A[i, j]$ равен 1, если человек i знаком с человеком j , $A[i, j] = A[j, i]$. Можно ли разбить людей на 2 группы, чтобы в каждой группе были только незнакомые люди.
40. Написать интерпретатор языка A

ОПИСАНИЕ ЯЗЫКА A

Программа на языке A - это последовательность операторов, каждый из которых записывается в отдельной строке. Пустые строки допускаются. Язык включает оператор присваивания, оператор перехода, условный оператор, оператор ввода, оператор вывода и оператор останова.

=====

Оператор присваивания:

ИМЯ:=ВЫРАЖЕНИЕ

здесь

ИМЯ - имя переменной, состоящее из одной латинской буквы, все переменные языка имеют целый тип и принимают значения от -1000 до 1000, переменные явно не описываются и по умолчанию имеют нулевые значения.

ВЫРАЖЕНИЕ - это целочисленное выражение, состоящее из операндов и знаков операций, операндами могут быть константы, имена переменных и указатели встроенных функций. Константами языка являются обычные целые десятичные константы от -1000 до 1000.

Язык содержит одну встроенную функцию `abs`, возвращающую абсолютную величину числового аргумента.

Язык включает 4 операции :

"+" -сложение,

"-" - вычитание,

"*" - умножение и

":"/" - целочисленное деление.

Приоритеты операций "*" и "/" выше, чем приоритеты операций "+" и "-", операции одного приоритета выполняются в последовательности слева направо, скобки в выражениях не допускаются.

=====

Оператор перехода:

go МЕТКА

здесь

МЕТКА - метка какого-либо оператора программы, представляющая собой целое число от 0 до 1000. Метка может быть указана в начале любого оператора и отделяется от оператора символом "#".

=====

Условный оператор:

if (ОТНОШЕНИЕ) ОПЕРАТОР

зесь

ОТНОШЕНИЕ - это конструкция состоящая из двух выражений, разделенных знаком отношения, знаки отношения - это "=", "!", ">", ">=", "<", "<=".

ОПЕРАТОР - это любой оператор кроме условного.

=====

Оператор ввода:

get(ИМЯ)

здесь

ИМЯ - имя переменной, вводимое значение набирается на клавиатуре в виде правильной константы языка А. Каждый новый оператор get начинается ввод с новой строки.

=====

Оператор вывода:

put(ВЫРАЖЕНИЕ/ЛИТЕРАЛ)

здесь ВЫРАЖЕНИЕ - правильное выражение языка А, ЛИТЕРАЛ - конструкция \$последовательность символов. При выводе

выражения на экран выводится константа, равная его значению, при выводе литерала на экран выводятся составляющие его символы без первого символа "\$". Каждый оператор put выводит с новой строки.

=====

Оператор останова:

finish

останавливает работу программы.