Анализ алгоритмов

22 января 2021 г.

Содержание

1 Скорость роста длины записи коэффициентов при реализации метода Гаусса (10)

$$\left\|b_{ij}^r\right\| \leq (r+1)M + r^2$$

- $||a||$ – длина записи числа a
- M – максимальная длина записи элементов матрицы
- r – ранг матрицы
- $||b_{ij}^r||$ – длина записи коэффицента после r-й итерации

2 Представление "длинного" числа в файле (массиве, списке) как числа в системе счисления по модулю p ($p = 1000$, если `integer` 2^{16}, если $p = 10000000$ `longinteger` 2^{32}). Запись из файла. Оценка числа шагов. Вывод в файл. Оценка числа шагов (14)

2.1 Представление длинного числа

Всякое целое неотрицательное число x может быть представлено в m-ичной системе счисления (при $m \geq 2$) в виде $x = m^{k-1}x_0 + m^{k-2}x_1 + \dots mx_{k-2} + x_{k-1}$. При этом k – длина записи m-ичного представления числа $x, 0 \leq x_i \leq m-1$ при $i = 0, \dots, k-1$

2.2 Запись из файла

Оценка числа шагов: квадратичное от длины записи исходного числа в файле количество "шагов".

Пусть в файле записано десятичное число, заданное словом $a_1 \cdots a_n$ $(0 \leq a_i \leq 9)$. Требуется представить его динамическим массивом (или списком).

Под "шагом" понимается одна из следуюших операций: считывание цифры из файла, запись цифры в целочисленный массив, выделение первой цифры многозначного числа и её удаление из него, приписывание цифры в конец числа. Заметим, что эти "шаги" не равнозначны, т.к. последние два требуют нахождения остатка от деления на 10, а также умножения на 10 и сложения.

2.3 Вывод в файл

Оценка числа шагов: линейное от длины записи исходного числа количество "шагов".

При выводе числа необходимо помнить, что в каждом элементе массива, в котором хранится многоразрядное число, записана не последовательность цифр, а число, записанное этими цифрами. Поэтому число, десятичная запись которого меньше, чем длина записи выбранного нами основания m, необходимо дополнить ведущими нулями.

Под "шагом" будем понимать одну из следующих операций: запись "макроцифры" в символьную переменную, сравнение длины записи "макроцифры" с $\|m-1\|$, дополнение строки ведущим нулём.

3 Сложение двух "длинных" положительных чисел. Оценка числа шагов (17)

Оценка числа шагов: общее число "шагов" при сложении двух неотрицательных чисел не превосходит $3\max\{A[0], B[0]\}+1$, то есть составляет $\mathrm{O}(\max\{A[0], B[0]\})$

Чтобы сложить два неотрицательных многоразрядных числа, записанных в массивы A и B, достаточно последовательно складывать по модулю m числа, записанные в $A[i], B[i]$ и $d[i]$ для $i=1,\ldots,\max\{A[0], B[0]\}$, где $d[1]=0$, при $i>1, d[i]-$ это 1 (если $A[i-1]+B[i-1]+d[i-1]>m$) или 0 в противном случае.

При подсчёте числа шагов в этом разделе под "шагом" понимается одна из следующих операций: вычисление $A[i-1]+B[i-1]+d[i-1] \mod m$, проверка условия $A[i-1]+B[i-1]+d[i-1]>m$ и вычисление $d[i]$

4 Предикаты равенства и неравенств "длинных" положительных чисел. Оценка числа шагов (18)

Оценка числа шагов: если под "шагом" понимать количество сравнений "макроцифр", то обшее число "шагов" такой процедуры не превосходит $A[0]$. В общем случае число "шагов" вычисления каждого из четырёх предикатов не превосходит $\min\{A[0], B[0]\}$.

Оценим число шагов вычисления значений предикатов $x=y$ и $x<y$ для случая, когда $A[0]=B[0]$.

Начиная со старшего разряда (то есть с $A[A[0]]$ и $B[B[0]]$) сравниваем значения чисел в $A[i]$ и $B[i]$ до тех пор, пока они совпадают. Если для некоторого $i_0 A\left[i_0\right] \neq B\left[i_0\right]$, то $x \neq y$. Если при этом $A\left[i_0\right]<B\left[i_0\right]$, то $x<y$, если $A\left[i_0\right]>B\left[i_0\right]$, то $x>y$.

5 Вычитание двух "длинных" положительных чисел. Оценка числа шагов (18)

Оценка числа шагов: общее число "шагов" при вычитании двух положительных чисел не превосходит $4\max\{A[0], B[0]\}+1$, то есть составляет $\mathrm{O}(\max\{A[0], B[0]\})$.

При подсчёте числа шагов в этом разделе под "шагом" понимается одна из следуюшци операций: вычисление $A[i-1]-B[i-1]-d[i-1] \pmod{m}$, проверка условия $A[i-1]+B[i-1]+d[i-1]>0$ и вычисление $d[i]$. Кроме того, предварительно проверяется условие $x \geq y$.

6 Умножение "длинного" числа на короткое. Оценка числа шагов (19)

Оценка числа шагов: общее число операций не превосходит $\max\{1, 2+6A[0]+\max\{1,3\}\} = 5+6A[0]$, то есть составляет $\mathrm{O}(A[0])$.

Здесь под шагом будем понимать одну из следуюших операций: умножение макроцифр, сложение макроцифр, вычисление неполного частного и остатка от деления результата предыдущих операций на m. В условном операторе после else выполняется одно присваивание и оператор цикла, в котором (помимо двух операций, необходимых для организации цикла) производятся: умножение, сложение, остатка от деления на m, вычисление неполного частного. Всего в операторе цикла 6 "шагов".

7 Умножение "длинных" чисел. Оценка числа шагов (20)

Оценка числа шагов: В предположении, что $B[0] \leq A[0]$ (это условие проверяется за 1 «шаг» и в противном случае можно умножать B на A), получаем оценку $O(A[0]B[0])$.

8 Деление "длинных" чисел. Оценка числа шагов (21)

Оценка числа шагов: $O(A[0]B[0] \cdot (A[0] - B[0]))$

Будем подбирать неполное частное от деления чисел x и y, записанных в массивах A и B, делением промежутка, в котором оно может находиться, пополам. Пусть L и U – нижняя и верхняя границы промежутка соответственно, $M = \left\lfloor \frac{L+U}{2} \right\rfloor$ – целая часть середины промежутка, $z = y \cdot M-$ число, которое будем сравнивать с делимым.

При этом будем предполагать, что число, записанное в A, больше числа, затисанного в B (в противном случае неполное частное равно 0, а остаток совпадает с делимым).

9 Оценки числа шагов метода Гауса при действиях с "длинными" числами

Итоговая асимптотика: $O(\min(n,m) \cdot nm)$

При $n = m$ эта оценка превращается в $O(n^3)$ Для длинных чисел получается $O(M * n^3)$.

10 Сортировки и оценки числа их шагов: Пузырёк. Сортировка вставками. Сортировка слияниями фон Неймана (25)

10.1 Пузырёк

Сложность: $O(n^2)$

В теле циклов сравниваются значения $a[i]$ и $a[j]$. В случае необходимости содержание элементов массива меняются местами. Обмен значениями переменных x и y можно осуществить с помошью трёх операторов присваивания с использованием вспомогательной переменной

$z : z := x; x := y; y := z$.[3] Таким образом, в теле цикла каждый раз выполняется не более четырёх операций.

10.2 Сортировка вставками

Сложность: $O(n^2)$

Элементы входной последовательности просматриваются по одному, и каждый новый поступивший элемент размещается в подходящее место среди ранее упорядоченных элементов.

10.3 Сортировка слияниями фон Неймана

Сложность: $O(n \log_2 n)$

Сортируемый массив разбивается на две части примерно одинакового размера; Каждая из получившихся частей сортируется отдельно, например — тем же самым алгоритмом; Два упорядоченных массива половинного размера соединяются в один.

11 Алгоритмы на графах, различные способы представления графа в компьютере (28)

- V – произвольное конечное множество;
- E – подмножество множества двуэлементных подмножеств множества V;
- $G = (V, E)$ – граф с множеством вершин V и множеством рёбер E;
- A – подмножество множества упорядоченных пар множества V;
- $G = (V, A)$ – орграф с множеством вершин V и множеством дуг A;
- n – количество вершин в графе;
- m – количество рёбер в графе;
- $N(v)$ – окружение вершины v, т.е. множество вершин, смежных с v;
- $OUT(v)$ – множество вершин орграфа, непосредственно достижимых из v;
- $IN(v)$ – множество вершин орграфа, из которых v непосредственно достижима;
- $\deg(v)$ – степень вершины v, т.е. количество вершин в окружении;
- w_{ij} – вес ребра $\{v_i, v_j\}$ или ребра (v_i, v_j) во взвешеном графе.

11.1 Матрица смежности

Матрица смежности графа – это квадратная матрица $A_{n\times n}$, элементы которой определены так:

$$a_{ij} = \begin{cases} 1 & \text{если } \{v_i, v_j\} \in E \\ 0 & \text{иначе} \end{cases}$$

11.2 Списки смежности

Списки смежности – это одномерный массив, i-ым элементом которого является список вершин, смежных с v_i, т.е. окружение вершшны v_i.

11.3 Матрица инцидентности

Матрица инцидентности графа – это матрица $B_{n\times m}$, элементы которой определены так:

$$b_{ij} = \begin{cases} 1 & \text{если } v_i \in e_j \\ 0 & \text{иначе} \end{cases}$$

12 Алгоритм поиска в глубину. Оценки числа шагов в зависимости от способа представления графа (28)

- **Матрица смежности:** $O\left(n^2\right)$ При обходе графа в глубину или в ширину для каждой вершины необходимо проверить все (при обходе в глубину - постепенно, а при обходе в ширину - сразу) вершины, смежные с данной. При использовании матрицы смежности для одной вершины это можно сделать за n проверок того, следует ли помешать вершины в стек или в очередь. Эта процедура выполняется для каждой из n вершин. Этим объясняется то, что алгоритмы, основанные на обходе графа в глубину или в ширину при представлении графа матрицей смежности, имеют оценку числа шагов вида $O\left(n^2\right)$.

  Для орграфа элементы матрицы смежности определяются так

  $$a_{ij} = \begin{cases} 1 & \text{если } (v_i, v_j) \in A \\ 0 & \text{иначе} \end{cases}$$

  Рассуждениями, аналогичными таковым для не ориентированного графа, получаем оценку числа шагов вида $O\left(n^2\right)$.

- **Списки смежности:** $O(n+m)$. Для графов с разными свойствами эта оценка может быть видоизменена.

  Если граф является дереном или лесом, то $m < n$ и оценка принимает вид $O(n)$.

  Ecru граф полный, то $m = \frac{n(n-1)}{2}$ и оценка принимает вид $O\left(n^2\right)$. Если степени всех вершин графа не превосходят некоторой константы C (существенно меньшей, чем n), то $m \leq Cn$ и оценка принимает вид $O(n)$.

  Если граф связен и степени вершин произвольны, то $m \geq n-1$ и оценка принимает вид $O(m)$.

  Для орграфа список смежности для вершины v состоит из вершин, входяших в $OUT(v)$ (или в $IN(v)$), Поскольку $\sum_{v\in V} \|OUT(v)\| = \sum_{v\in V} \|IN(v)\| = m$, то рассуждениями, аналогичными для не ориентированного графа, получаем оценку числа шагов вида $O(n+m)$.

- **Матрица инцидентности:** $O(nm)$

13 Алгоритм поиска в ширину. Оценки числа шагов в зависимости от способа представления графа (28)

- **Матрица смежности:** $O\left(n^2\right)$ При обходе графа в глубину или в ширину для каждой вершины необходимо проверить все (при обходе в глубину - постепенно, а при обходе в ширину - сразу) вершины, смежные с данной. При использовании матрицы смежности для одной вершины это можно сделать за n проверок того, следует ли помешать вершины в стек или в очередь. Эта процедура выполняется для каждой из n вершин. Этим объясняется то, что алгоритмы, основанные на обходе графа в глубину или в ширину при представлении графа матрицей смежности, имеют оценку числа шагов вида $O\left(n^2\right)$.

  Для орграфа элементы матрицы смежности определяются так

  $$a_{ij} = \begin{cases} 1 & \text{если } (v_i, v_j) \in A \\ 0 & \text{иначе} \end{cases}$$

  Рассуждениями, аналогичными таковым для не ориентированного графа, получаем оценку числа шагов вида $O\left(n^2\right)$.

- **Списки смежности:** $O(n+m)$. Для графов с разными свойствами эта оценка может быть видоизменена.

  Если граф является дереном или лесом, то $m < n$ и оценка принимает вид $O(n)$.

  Ecru граф полный, то $m = \frac{n(n-1)}{2}$ и оценка принимает вид $O\left(n^2\right)$. Если степени всех вершин графа не превосходят некоторой константы C (существенно меньшей, чем n), то $m \leq Cn$ и оценка принимает вид $O(n)$.

  Если граф связен и степени вершин произвольны, то $m \geq n-1$ и оценка принимает вид $O(m)$.

  Для орграфа список смежности для вершины v состоит из вершин, входяших в $OUT(v)$ (или в $IN(v)$), Поскольку $\sum_{v\in V} \|OUT(v)\| = \sum_{v\in V} \|IN(v)\| = m$, то рассуждениями, аналогичными для не ориентированного графа, получаем оценку числа шагов вида $O(n+m)$.

- **Матрица инцидентности:** $O(nm)$

14 Задачи, решаемые с помощью этих алгоритмов: выделение компонент связности; проверка на двудольность и выделение долей; выделение остова графа

14.1 Выделение компонент связности

Компонента связности графа G – максимальный (по включению) связный подграф графа G. Другими словами, это подграф $G(U)$, порождённый множеством $U \subseteq V(G)$ вершин, в котором для любой пары вершин $u, v \in U$ в графе G существует (u, v)-цепь и для любой пары вершин $u \in U, w \notin U$ не существует (u, w)-цепи.

Для выделения компонент связности можно использовать поиск в ширину или поиск в глубину. При этом затраченное время будет **линейным** от суммы числа вершин и числа рёбер графа.

14.2 Проверка на двудольность и выделение долей

Двудольный граф или биграф в теории графов – это граф, множество вершин которого можно разбить на две части таким образом, что каждое ребро графа соединяет какую-то вершину из одной части с какой-то вершиной другой части, то есть не существует рёбер между вершинами одной и той же части.

14.3 Выделение остова графа

15 Нахождение остова минимального веса. Метод Р. Прима. Оценки числа шагов

16 Алгоритм Дейкстры поиска кратчайшего пути. Оценки числа шагов

17 Нахождение циклов и мостов в графе. Оценки числа шагов

18 Эйлеров цикл. Оценки числа шагов

19 Гамильтонов цикл. Оценки числа шагов

20 Алгоритм генерации всех независимых множеств. Оценки числа шагов

21 Теорема о НМ, ВП, КЛИКА. Оценки числа шагов

22 Отличия между интуитивным и математическим понятиями

23 Машины Тьюринга и их модификации. Тезис Тьюринга-Чёрча

24 Теорема о числе шагов МТ, моделирующей работу k-ленточной МТ

25 Недетерминированные МТ. Теорема о числе шагов МТ, моделирующей работу недетерминированной МТ

26 Понятия сложности алгоритма от данных, сложность алгоритма, сложность задачи. Верхняя и нижняя оценки сложности

27 Соотношение между временем работы алгоритма требуемой памятью

28 Классы алгоритмов и задач. Схема обозначений

29 Классы P, NP и $P-SPACE$. Соотношения между этими классами

30 Полиномиальная сводимость и полиномиальная эквивалентность

31 Полиномиальная сводимость задачи ГЦ к задаче КОМИВОЯЖЁР

32 Классы эквивалентности по отношению полиномиальной