Анализ алгоритмов

Ульянов Михаил Васильевич

2019

Оглавление

Глава 1

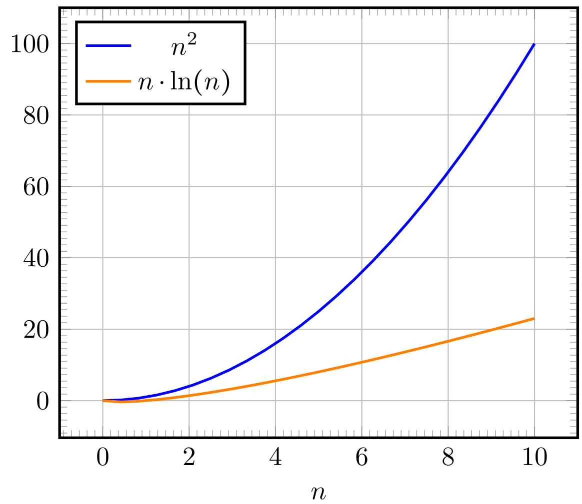
Исторический очерк

1. **1900**
   Д. Гильберт - 23 проблемы 1931 - К. Гедель доказал теорему о неполноте

2. **1936**
   А.Тьюринг, Э.Л.Пост - Теория алгоритмов (начало)

   - формализация понятия
   - общие свойства
   - обнаружение алгоритмически неразрешимых задач

3. **1960е**
   Теория сложности вычислений NPC
   $O(n^2)$ $O(n \cdot \ln(n))$



4. **Начало 1970х**
   Практический анализ алгоритмов Д.Э. Кнут

Глава 2

Схема выбора алгоритмического обеспечения

Нет:

A. Новый (метод разработки)

B. Комбинированные элементы $(A_1 + A_2 + A_3)$

$$Q(q_1, ..., q_m) = \sum \alpha_i q_i \to R^1 \text{ - комплексные оценки}$$

$$\begin{cases} a_{11}x_1 + a_{12}x_2 = b_1 \\ a_{21}x_1 + a_{22}x_2 = b_2 \end{cases}$$

$$a + ib = (c + id) =^{det} (ac - bd) + i(bc + ad)$$

Глава 3

1936 год Э.Л. Пост "Финитные комбинаторные процессы - формулировка 1"

3.1 Терминология

Общая проблема = задача
Конкретная проблема = индивидуальная задача

3.2 Общая проблема

Общая → множество всех конкретных
Решение общей → решение каждой конкретной

3.3 Пространство символов

←							→

	Помечен, не помечен	

Конкретная проблема задается внешней силой путем пометки конечного числа символов.

3.4 Работник (процессор)

1. → (R)
2. ← (L)
3. ✓ Поставить метку, если пусто
4. ξ Стереть, если есть
5. ? $\overset{\text{да}}{\to} N^o$ строки
   $\overset{\text{нет}}{\to} N^o$ строки
6. stop

3.5 Примеры

	✓	✓	✓	✓	...	✓	

1. ξ
2. →

3. $?\begin{array}{l}\xrightarrow{\text{да}} 1\\ \xrightarrow{\text{нет}} 4\end{array}$

4. stop

3.6 Гипотеза Поста

a) Программа применима к общей, если ∀ конкретной нет коллизий в операциях 3,4

b) программа заканчивается, если stop

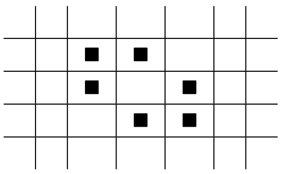
c) Если ∀ конкретной внеш сила распознает правильный ответ, то Ф1П есть 1-решение общей

d) Мы вправе рассматривать все более и более широкие формулы

пространство символов	
алфавита	логически сводимы к формуле 1
набор инструкций	

a + b = Финитный 1 процесс

3.7 1984 Муравей Лэнгтона



$$W \to (B, R)$$

$$B \to (W, L)$$

Глава 4

Терминология

4.1 Вход алгоритма

Конкретная проблема $\Rightarrow$ индивидуальная задача $\Rightarrow$ Вход $D = \{d_i | i = 1, m\}$

4.2 Длина входа

$n \to |D|$?

$$\mu_z(D) \to g(|D|)$$

$$\mu_{sort}(D) = |D| - 1$$

Умножение матриц $n \times n$

$$|D| = 2n^2 + 1$$

$$\mu_z(D) = n = \sqrt{\frac{|D| - 1}{2}}$$

4.3 Трудоемкость

$f_A(D)$ - число элементарных операций принятой модели вычислений заданных алгоритмом A на входе D

$$f_{A_1}(D) < F_{A_2}(D)$$

4.4 D_n

$$D_n = \{D | \mu_z(D) = n\}$$

$$\beta = 16 \quad |D_{10}| = 2^{160}$$

4.5 Переход к n

$$f_A^{\vee}(n) \stackrel{def}{=} \min_{D \in D_n} f_A(D) \text{ - лучший случай}$$

$$f_A^{\wedge}(n) \stackrel{def}{=} \max_{D \in D_m} f_A(D) \text{ - худший случай}$$

$$\overline{f}_A(n) = \sum_{D \in D_n} p(D) - f_A(D)$$

$$p(D) = \frac{1}{|D_n}$$

4.6 Память

$$V(D) = \underbrace{|D|}_{\text{Вход}} + \underbrace{|R|}_{\text{Результат}} + V_{\text{допольнительно памяти}}(D) + V_{\text{программы}}$$

$$V_A(D) = V_{\text{доп}}(D)|V^{\vee}(n), V^{\vee}(n), ...$$

Глава 5

Примеры

5.1 Умножение матриц

```
MultM(A,B,n;C)
    for i ← 1 to n
        for j ← 1 to n
            s ← 0
            for k ← 1 to n
                s ← s + A[i,k] * B[k,j]
            c[i,j] ← s
End
```

$$f_A^{\vee}(n) = f_A^{\wedge}(n) = f_A(n) = 1 + n(3 + 1 + n(3 + 2 + n(3 + 7) + 3)) = 10n^3 + 8n^2 + 4n + 1$$

5.2 Max

```
Max(A,n;M)
    M ← A[1]
    for j ← 2 to n
        if M < A[i]
            M ← A[i]
End
```

$$f_A^{\vee}(n) = 2 + 1 + (n - 1)(3 + 2) = 5n - 2$$

$$f_A^{\wedge}(n) = 2 + 1 + (n_1)(3 + 2 + 2) = 7n - 4$$