Алгоритмы и структуры данных-1

Подготовка к контрольной работе. Повторение материала

Практическое занятие 6 — 07.10–12.10.2024

2024-2025 учебный год

ПЛАН

Инвариант цикла — инструмент анализа корректности

Асимптотические оценки временной сложности
Символы Ω, O, Θ — определения и их свойства

Алгоритмы «разделяй (уменьшай)–и–властвуй»
Рекуррентные соотношения

ИНВАРИАНТ ЦИКЛА

- Сформулируйте общее понятие инварианта цикла

INIT — инвариант верен до входа в цикл
начальные условия

MNT — инвариант верен во время каждой итерации цикла
промежуточные вычисления

TRM — инвариант верен при выходе из цикла
результат вычислений

Упражнение 1 **Инвариант алгоритма**

```
algorithm(A, n)
    B — массив для хранения n чисел
    B[0] = A[0]

    for i = 0 to n - 1
        if A[i] > B[i - 1]
            B[i] = A[i]
        else
            B[i] = B[i - 1]

    return B[n - 1]
```

1. Сформулируйте условие P, которое подходит в качестве инварианта данного алгоритма для обработки целочисленного массива A размера n.
2. Выполните проверку предложенного инварианта.

АСИМПТОТИЧЕСКИЕ ОЦЕНКИ

- Чем точная функция $T(n)$ временной сложности отличается от ее асимптотической оценки?

1. Если $T(n) = \Theta(f(n))$, то ...
2. Если $T(n) = O(f(n))$, то ...
3. Если $T(n) = \Omega(f(n))$, то ...

Упражнение 2 **Асимптотически нотации**

Подтвердите или опровергните истинность равенств.

- $O(1) \cdot O(n^2) + \Theta(n) = O(n^3)$
- $O(1) \cdot O(1) \cdot \ldots \cdot O(1) + O(1) = O(1)$
- $\Theta(n^2) + O(n) = \Omega(n)$
- $\Theta(n^3 - n^2) = \Omega(n^3)$
- $n^2 + n \cdot \log_2 n - 4 = \Omega(n \cdot \log n)$

Упражнение 3 **Оценка сложности**

```
algorithm(n)
    x = 0
    i = n

    while i >= 2
        j = pow(n, 0.25) * i

        while j >= i
            x = x + 1
            j = j - 10

        i = i / sqrt(n)

    return x
```

Определите асимптотическую точную границу временной сложности данного алгоритма, на вход которого подается целое число $n \geq 2$.

РЕКУРРЕНТНОЕ СООТНОШЕНИЕ

$$T(n) = a \cdot T(n - b) + O\left(n^k \cdot f(n)\right)$$

$$T(n) = a \cdot T\left({}^{n}/_{b}\right) + O\left(n^k \cdot f(n)\right)$$

- Могут ли парамеры a, b зависеть от n?
- Определите ограничения применимости метода подстановки.
- Дерево рекурсии — это …
- Определите ограничения применимости мастер–теорем.

Упражнение 4 **Асимптотика рекурсии**

```
algorithm(A, n)
    if n <= 20
        return A[1]

    j = 3
    while j < n
        A[j] = A[j - 1] + A[j]
        j = j + sqrt(n)

    y = algorithm(A, n - 3)

    return y
```

1. Составьте рекуррентное соотношение, описывающее сложность данного алгоритма обработки массива $A[0..n]$.
2. Вычислите асимптотическую верхнюю границу сложности с помощью
   - дерева рекурсии
   - мастер–теоремы, если она применима