A1. Временная сложность рекурсии

Демченко Георгий Павлович , БПИ-235

Условие

algorithm1	algorithm2
`` 1 algorithm1(A, n) 2 if n <= 20 3 return A[n] 4 x = algorithm1(A, n - 5) 5 6 for i = 1 to ⌊n / 2⌋ 7 for j = 1 to ⌊n / 2⌋ 8 A[i]= A[i] - A[j] 9 x = x + algorithm1(A, n - 8) 10 11 return x ```	``` 1 algorithm2(A, n): 2 if n <= 50 3 return A[n] 4 x = algorithm2(A, ⌊n / 4⌋) 5 6 for i = 1 to ⌊n / 3⌋ 7 A[i] = A[n - i] - A[i] 8 9 x = x + algorithm2(A, ⌊n / 4⌋) 10 11 return x ```

algorithm1:

```
1  algorithm1(A, n)
2      if n <= 20
3          return A[n]
4      x = algorithm1(A, n - 5)
5
6      for i = 1 to ⌊n / 2⌋
7          for j = 1 to ⌊n / 2⌋
8              A[i]= A[i] - A[j]
9      x = x + algorithm1(A, n - 8)
10
11     return x
```

algorithm2:

```
1  algorithm2(A, n):
2      if n <= 50
3          return A[n]
4      x = algorithm2(A, ⌊n / 4⌋)
5
6      for i = 1 to ⌊n / 3⌋
7        A[i] = A[n - i] - A[i]
8
9      x = x + algorithm2(A, ⌊n / 4⌋)
10
11     return x
```

1. Временная сложность алгоритмов

Под "to m" будем воспринимать $\leq m$

"Предполагается, что все арифметические операции выполняются за постоянное время" - будем считать, что все арифм. операции выполняются за c_1

Все сравнения выполняются за c_2

Все инициализации/присвоения/return выполняются за c_3

algorithm1

Номер строки	Количество итераций	Затраты
2-3	1	c_2 (**return** невозможно отследить, опустим)
4	1	$T(n-5)+c_3+c_1$ - присвоение и вычитание
6	1	c_3 - инициализация

Номер строки	Количество итераций	Затраты
6	$\lfloor \frac{n}{2} \rfloor + 1$	c_2 - сравнение
6	$\lfloor \frac{n}{2} \rfloor$	c_1 - инкремент
7	$\lfloor \frac{n}{2} \rfloor$	c_3 - инициализация
7	$\lfloor \frac{n}{2} \rfloor^2 + \lfloor \frac{n}{2} \rfloor$	c_2 - сравнение
7	$\lfloor \frac{n}{2} \rfloor^2$	c_1 - инкремент
8	$\lfloor \frac{n}{2} \rfloor^2$	$c_3 + \mathrm{c}_1$ - присвоение и вычитание
9	1	$T(n-8) + c_3 + 2c_1$ - присвоение, сумма и вычитание
11	1	c_3

$\Rightarrow T(n) = T(n-5) + T(n-8) + \lfloor \frac{n}{2} \rfloor \cdot (c_2 + c_1 + c_3 + c_2) + \lfloor \frac{n}{2} \rfloor^2 \cdot (c_2 + c_1 + c_3 + c_1) + (c_2 + c_2 + c_3 + c_3 + c_1 + 2c_1 + c_3) =$

$= T(n-5) + T(n-8) + \lfloor \frac{n}{2} \rfloor \cdot (2c_2 + c_1 + c_3) + \lfloor \frac{n}{2} \rfloor^2 \cdot (c_2 + 2c_1 + c_3) + (2c_2 + 3c_3 + 3c_1)$

Проведем замены констант, для красвого вида

- $2c_2 + c_1 + c_3 = c_4$
- $c_2 + 2c_1 + c_3 = c_5$
- $2c_2 + 3c_3 + 3c_1 = c_6$

$\Rightarrow T(n) = T(n-5) + T(n-8) + \lfloor \frac{n}{2} \rfloor^2 \cdot c_5 + \lfloor \frac{n}{2} \rfloor \cdot c_4 + c_6$

algorithm2

Номер строки	Количество итераций	Затраты
2-3	1	c_2 (**return** невозможно отследить, опустим)
4	1	$T(\lfloor \frac{n}{4} \rfloor) + c_3 + c_1$ - присвоение и деление
6	1	c_3

Номер строки	Количество итераций	Затраты
6	$\lfloor \frac{n}{3} \rfloor + 1$	c_2 - сравнение
6	$\lfloor \frac{n}{3} \rfloor$	c_1 - инкремент
7	$\lfloor \frac{n}{3} \rfloor$	$c_3 + c_1$ - присвоение и вычитание
9	1	$T(\lfloor \frac{n}{4} \rfloor) + c_3 + 2c_1$ - присвоение, сумма и деление
11	1	c_3

$\Rightarrow T(n) = 2 \cdot T(\lfloor \frac{n}{4} \rfloor) + \lfloor \frac{n}{3} \rfloor \cdot (2c_1 + c_2 + c_3) + (2c_2 + 3c_1 + 4c_3)$

Проведем замены констант, для красвого вида

- $2c_1 + c_2 + c_3 = c_4$
- $2c_2 + 3c_1 + 4c_3 = c_5$

$\Rightarrow T(n) = 2 \cdot T(\lfloor \frac{n}{4} \rfloor) + \lfloor \frac{n}{3} \rfloor \cdot c_4 + c_5$

2. $T(n) = \Theta(f(n))$

algorithm2

$T(n) = 2 \cdot T(\lfloor \frac{n}{4} \rfloor) + \lfloor \frac{n}{3} \rfloor \cdot c_4 + c_5$

Предположение

Пусть $f(n) = n, \text{т.е } T(n) = \Theta(n)$

$\Rightarrow \exists\, d_1, d_2 \in \mathbb{R}^+ : \forall \mathbb{N} > N_0 : d_2 \cdot n \leq T(n) \leq d_1 \cdot n$

- Пусть $N_0 = 1$

Ограничение сверху

- Для точности ограничения сверху раскроем $\lfloor \frac{n}{4} \rfloor$ и $\lfloor \frac{n}{3} \rfloor$ как $\frac{n}{4}$ и $\frac{n}{3}$, так как $\lfloor \frac{n}{k} \rfloor \leq \frac{n}{k}\ \forall k \in \mathbb{N}$

$T(n) \leq d_1 \cdot n$

$\Rightarrow T(n) = 2 \cdot T(\lfloor \frac{n}{4} \rfloor) + \lfloor \frac{n}{3} \rfloor \cdot c_4 + c_5 \leq 2 \cdot T(\frac{n}{4}) + \frac{n}{3} \cdot c_4 + c_5 \leq 2 \cdot d_1 \cdot \frac{n}{4} + \frac{n}{3} \cdot c_4 + c_5 \leq d_1 \cdot n$

$d_1 \cdot \frac{n}{2} + \frac{n}{3} \cdot c_4 + c_5 \leq d_1 \cdot n$

$\frac{n}{3} \cdot c_4 + c_5 \leq d_1 \cdot \frac{n}{2}$

$d_1 \geq \frac{2 \cdot c_4}{3} + \frac{2 \cdot c_5}{n} \ \forall \mathbb{N} \geq 2$

$\Rightarrow d_1 \geq \frac{2 \cdot c_4}{3} + c_5 \in \mathbb{R}^+$ - Предположение о верхней границе верно

Ограничение снизу

- Для точности ограничения снизу раскроем $\lfloor \frac{n}{4} \rfloor$ и $\lfloor \frac{n}{3} \rfloor$ как $(\frac{n}{4} - 1)$ и $(\frac{n}{3} - 1)$, так как $\frac{n}{k} - 1 < \lfloor \frac{n}{k} \rfloor \ \forall k \in \mathbb{N}$

$d_2 \cdot n \leq T(n)$

$\Rightarrow d_2 \cdot n \leq 2 \cdot (\frac{n}{4} - 1) \cdot d_2 + (\frac{n}{3} - 1) \cdot c_4 + c_5 \leq 2 \cdot T(\frac{n}{4} - 1) + (\frac{n}{3} - 1) \cdot c_4 + c_5 \leq 2 \cdot T(\lfloor \frac{n}{4} \rfloor) + \lfloor \frac{n}{3} \rfloor \cdot c_4 + c_5 = T(n)$

$d_2 \cdot n \leq \frac{n}{2} \cdot d_2 - 2 \cdot d_2 + \frac{n}{3} \cdot c_4 + c_5 - c_4$

$d_2 \cdot (\frac{n+4}{2}) \leq \frac{n}{3} \cdot c_4 + c_5 - c_4$

Вспомним про произведенные замены

$c_5 - c_4 = 2c_2 + 3c_1 + 4c_3 - (2c_1 + c_2 + c_3) = c_1 + c_2 + 3c_3 = c_6 > 0$

$\Rightarrow d_2 \cdot (\frac{n+4}{2}) \leq \frac{n}{3} \cdot c_4 + c_6$

(т.к $N_0 = 1$) $\forall \mathbb{N} \geq 2:$

$\Rightarrow d_2 \cdot (\frac{n+4}{2}) \leq d_2 \cdot \frac{5n}{2} \leq \frac{n}{3} \cdot c_4 + c_6$

$d_2 \leq \frac{2}{15} \cdot c_4 + \frac{2 \cdot c_6}{5n}$

$\Rightarrow d_2 \leq \frac{2}{15} \cdot c_4 \in \mathbb{R}^+$ - Предположение о нижней границе верно

Значит изначальное предположение верно и $T(n) = \Theta(n)$

algorithm1

$T(n) = T(n-5) + T(n-8) + \lfloor \frac{n}{2} \rfloor^2 \cdot c_5 + \lfloor \frac{n}{2} \rfloor \cdot c_4 + c_6$

Невозможно сформировать ассимтотически-точную границу T(n) из-за неявной глубины рекурсии. Будем отдельно рассматривать верхнюю и нижнюю границы

Ограничение сверху

- Для точности ограничения сверху раскроем $\lfloor \frac{n}{2} \rfloor$ как $\frac{n}{2}$ так как $\lfloor \frac{n}{k} \rfloor \leq \frac{n}{k} \; \forall k \in \mathbb{N}$
- Также заменим $T(n-5) + T(n-8)$ на $2 \cdot T(n-5)$, т.к тогда мы имеем наибольшую глубину рекурсии и наибольшие затраты на подзадачи на уровне рекурсии

$\Rightarrow T(n) \leq 2 \cdot T(n-5) + (\frac{n}{2})^2 \cdot c_5 + \frac{n}{2} \cdot c_4 + c_6 = 2 \cdot T(n-5) + O(n^2)$

$\Rightarrow T(n) \leq 2 \cdot T(n-5) + r_1 \cdot n^2 \quad \forall \mathbb{N} > N_1$ - некоторый N_1 для $O(n^2)$

Докажем, что $T(n) = O(2^{\frac{n}{5}} \cdot n^2)$

$\Rightarrow \exists \, d_1 \in \mathbb{R}^+ : \forall \mathbb{N} > N_0 : T(n) \leq d_1 \cdot 2^{\frac{n}{5}} \cdot n^2$

Пусть $N_0 = max(N_1; 9)$

$T(n) \leq 2 \cdot T(n-5) + r_1 \cdot n^2 \leq d_1 \cdot 2^{\frac{n}{5}} \cdot n^2$

$\Rightarrow 2 \cdot T(n-5) + r_1 \cdot n^2 \leq 2 \cdot d_1 \cdot 2^{\frac{n-5}{5}} \cdot (n-5)^2 + r_1 \cdot n^2 \leq d_1 \cdot 2^{\frac{n}{5}} \cdot n^2$

$d_1 \cdot (2^{\frac{n}{5}} \cdot n^2 - 2 \cdot 2^{\frac{n-5}{5}} \cdot (n-5)^2) \geq r_1 \cdot n^2$

$d_1 \cdot (2^{\frac{n}{5}} \cdot n^2 - 2 \cdot \frac{2^{\frac{n}{5}}}{2} \cdot (n^2 - 10n + 25)) \geq r_1 \cdot n^2$

$d_1 \cdot (2^{\frac{n}{5}} \cdot (n^2 - n^2 + 10n - 25)) \geq r_1 \cdot n^2$

$d_1 \cdot 2^{\frac{n}{5}} \cdot (10n - 25) \geq r_1 \cdot n^2$

$\forall \mathbb{N} > N_0 : (10n - 25) > n$

$\Rightarrow d_1 \cdot 2^{\frac{n}{5}} \cdot (10n - 25) \geq d_1 \cdot 2^{\frac{n}{5}} \cdot n \geq r_1 \cdot n^2$

$d_1 \geq \frac{r_1 \cdot n}{2^{\frac{n}{5}}}$

$\Rightarrow d_1 \geq \frac{5 \cdot r_1}{2} \; \in \mathbb{R}^+$

$\Rightarrow T(n) = O(2^{\frac{n}{5}} \cdot n^2)$ - **верхняя граница**

Ограничение снизу

- Для точности ограничения сверху раскроем $\lfloor \frac{n}{2} \rfloor$ как $(\frac{n}{2} - 1)$ так как $\frac{n}{k} - 1 \leq \lfloor \frac{n}{k} \rfloor \; \forall k \in \mathbb{N}$

- Также заменим $T(n-5)+T(n-8)$ на $2\cdot T(n-8)$, т.к тогда мы имеем наименьшую глубину рекурсии и наименьшие затраты на подзадачи на уровне рекурсии

$\Rightarrow 2\cdot T(n-8)+(\frac{n}{2}-1)^2\cdot c_5+(\frac{n}{2}-1)\cdot c_4+c_6 \leq T(n)=T(n-5)+T(n-8)+\lfloor\frac{n}{2}\rfloor^2\cdot c_5+\lfloor\frac{n}{2}\rfloor\cdot c_4+c_6$

$2\cdot T(n-8)+(\frac{n}{2}-1)^2\cdot c_5+(\frac{n}{2}-1)\cdot c_4+c_6=2\cdot T(n-8)+(\frac{n^2}{4}-n+1)\cdot c_5+(\frac{n}{2}-1)\cdot c_4+c_6=$

$=2\cdot T(n-8)+n^2\cdot\frac{c_5}{4}+n\cdot(\frac{c_4}{2}-c_5)+(c_5-c_4+c_6)$

Произведем очередные замены для красоты:

- $\frac{c_5}{4}=c_7$
- $(\frac{c_4}{2}-c_5)=c_8$
- $(c_5-c_4+c_6)=c_9$

$\Rightarrow 2\cdot T(n-8)+n^2\cdot c_7+n\cdot c_8+c_9\leq T(n)$

Очевидно, что $n^2\cdot c_7+n\cdot c_8+c_9=\Omega(n^2)$, можно подробно расписать, но думаю факт достаточно очевидный, получится $d_2\leq c_7\in\mathbb{R}^+\ \forall\mathbb{N}>N_1$

$\forall\mathbb{N}>N_1$

$\Rightarrow 2\cdot T(n-8)+d_2\cdot n^2\leq 2\cdot T(n-8)+n^2\cdot c_7+n\cdot c_8+c_9\leq T(n)$

- **Можно расписать до дереву рекурсии и получить $\sum_{i=0}^{[\frac{n}{8}]-2}(2^i\cdot d_2(n-8i)^2)$, что достаточно тяжело оценить снизу из-за слагаемых в сумме (приедтся сильно ограничивать слагаемые, вплоть до константы, что ударить по оценке), либо полноценно раскрывать и считать 3 отдельные суммы.**
- **Иначе можно сделать менее точное предположение $T(n)=\Omega(n^2)$ и доказать его**

$\Rightarrow \exists\, d_1\in\mathbb{R}^+:\forall\mathbb{N}>N_0: d_1\cdot n^2\leq T(n)$

Пусть $N_0=max(N_1,20)$

$\Rightarrow d_1\cdot n^2\leq 2\cdot T(n-8)+d_2\cdot n^2\leq T(n)$

$d_1\cdot n^2\leq 2\cdot d_1\cdot(n-8)^2+d_2\cdot n^2\leq 2\cdot T(n-8)+d_2\cdot n^2$

$d_1\cdot(n^2-2(n^2-16n+64))\leq d_2\cdot n^2$

$d_1\cdot(32n-n^2+128)\leq d_2\cdot n^2$

$d_1 \cdot (32n - n^2 + 128) \leq d_1 \cdot n^2 \leq d_2 \cdot n^2$

$d_1 \leq d_2 \in \mathbb{R}^+$

$\Rightarrow T(n) = \Omega(n^2)$ - **нижняя грацница**