Основные алгоритмы. Домашняя работа 1 неделя

Зайнуллин Амир

8 февраля 2023 г.

Задача №1

1. Алгоритм выведет последовательность простых чисел от 2 до n. При каждой процедуре алгоритм записывает 1 в каждую ячейку, индекс которой кратен k. Следовательно, следующие процедуры будут игнорировать данные ячейки. Т.к $k > 1$ то значит будет игнорировать все составные индексы, тогда будет выводит все простые индексы.

2. Рассмотрим одну процедуру. Сначала алгоритм доходит до первой нулевой ячейки от 2 до k индекса. Далее алгоритм идет по массиву дальше с шагом один и через каждые k клеток записывает в ячейку единицу Т.е за одну процедуру алгоритм обходит весь массив размером $n - 1$. Количество выполненных процедур - сколько простых чисел вывел. Допустим их количество p. Тогда временная сложность равна $(n-1) \cdot p$. Выполняется $1 \leq p \leq n$. Тогда оценки получаются равными $\Omega(n)$ и $O(n^2)$.

3. Да, т.к временная сложность $O(n^2)$.

Задача №2

1. Допустим, да. Тогда $\exists C > 0, \exists n_0 \in \mathbb{N} : \forall n > n_0 \quad n \leqslant C \cdot nlogn$
   $\exists N \in \mathbb{N} : \forall n > N \quad 1 \leqslant log(n)$ Знаем, что такое N существует
   $\exists N \in \mathbb{N} : \forall n > N \quad n \leqslant nlog(n)$
   $\exists C = 1 : \forall n > N \quad n \leqslant nlog(n)$
   Верно

2. Допустим, да. Тогда $\exists C > 0, \exists n_0 \in \mathbb{N} : \forall n > n_0 \quad nlog(n) \geqslant C \cdot n^{1+\varepsilon}$
   $\exists C > 0, \exists n_0 \in \mathbb{N} : \forall n > n_0 \quad \dfrac{log(n)}{C} \geqslant n^{\varepsilon}$
   $\exists C > 0 : \forall n > 1 \quad ln\left[\dfrac{log(n)}{C}\right] \geqslant \varepsilon \cdot ln(n)$
   $\varepsilon \leqslant ln\left[\dfrac{log(n)}{C}\right] \cdot \dfrac{1}{ln(n)}$
   Видно, что числитель возрастает медленне чем знаменатель, т.к числитель порядка ln(log(n)), а знаменатель ln(n)). Значит правая часть будет убывать. Тогда неверно.

Задача №3

$$f(n) = O(n^2),\ g(n) = \Omega(1),\ g(n) = O(n),\ h(n) = \frac{f(n)}{g(n)}$$

$\exists C_1 > 0, \exists N_1 \in \mathbb{N} : \forall n > N_1 \quad f(n) \leqslant C_1 \cdot n^2$
$\exists C_2 > 0, \exists N_2 \in \mathbb{N} : \forall n > N_2 \quad g(n) \geqslant C_2$
$\exists C_3 > 0, \exists N_3 \in \mathbb{N} : \forall n > N_3 \quad g(n) \leqslant C_3 \cdot n$

1. Если $f(n) = nlogn \quad g(n) = 1$
   $h(n) = nlogn = \Theta(nlog(n))$
   Значит такое возможно
2. $h(n) = \dfrac{f(n)}{g(n)} \leqslant \dfrac{C_1 n^2}{C_2}$ Значит $h(n) = O(n^2)$. Значит пункт б невозможен
3. Верхнюю оценку мы дали. Нижняя оценку не можем, тк не знаем нижней оценки для $f(n)$.

Задача №4

$$\sum_{i=1}^{n} i^{3/2} \leqslant \sum_{i=1}^{n} \sqrt{i^3 + 2i + 5} \leqslant \sum_{i=1}^{n} \sqrt{i^3 + 6i^3 + 12i + 8}$$

$$\sum_{i=1}^{n} i^{3/2} \leqslant \sum_{i=1}^{n} \sqrt{i^3 + 2i + 5} \leqslant \sum_{i=1}^{n} \sqrt{(i+2)^3}$$

$$\sum_{i=1}^{n} i^{3/2} \leqslant \sum_{i=1}^{n} \sqrt{i^3 + 2i + 5} \leqslant \sum_{i=1}^{n} (i+2)^{3/2}$$

Слагаемое слева из семинарской задачи равно $\Theta(n^{5/2})$. Аналогично справа $\Theta((n+2)^{5/2}) = \Theta(n^{5/2})$ Тогда ответ $\Theta(n^{5/2})$.

Задача №5

$$g(n) = \Theta(n^{100}),\ \text{значит}\ \exists C_1,\ C2 > 0, \exists N \in \mathbb{N} : \forall n > N \quad C_1 \cdot n^{100} \leqslant g(n) \leqslant C_2 \cdot n^{100}$$

$$(3 + o(1))^n + C_1 n^{100} \leqslant (3 + o(1))^n + \Theta(n^{100}) \leqslant (3 + o(1))^n + C_2 n^{100}$$

$$log\left[(3 + o(1))^n\right] = n \cdot (3 + o(1))$$

$$log\left(Cn^{100}\right) = 100 \cdot log(Cn)$$

Так как $log(n) = O(n)$, то вторым слагаемым можно будет пренебречь. $\log f(n) = n \cdot (3 + o(1)) = \Theta(n)$ Верно

Задача №7

$$7 \equiv 7 \quad mod(167)$$
$$7^3 \equiv 9 \quad mod(167)$$
$$7^6 \equiv 81 \quad mod(167)$$
$$7^{12} \equiv 48 \quad mod(167)$$
$$7 \cdot 7^{12} \equiv 2 \quad mod(167)$$

Ответ: 2

Задача №8

1. $T_1(1) = T_1(2) = T_1(3) = 1$
   $T_2(1) = T_2(2) = T_2(3) = 1$
   $T_1(n) = T_1(n-1) + cn$
   $T_1(4) = T_1(3) + 4c = 1 + 4c$
   $T_1(5) = T_1(4) + 5c = 1 + 4c + 5c$
   $T_1(n) = 1 + 4c + 5c + \ldots + cn = 1 + c(4 + 5 + \ldots + n)$
   $= 1 + c \cdot \left[\frac{n(n+1)}{2} - 1 - 2 - 3\right] = \theta(n^2)$

2.

$$\begin{aligned}
&T_2(n) = T_1(n-1) + 4T_2(n-3)\\
&k^3 = k^2 + 4\\
&k^3 - k^2 - 4 = 0\\
&(k-2)\left(k^2 + k + 2\right) = 0\\
&k = 2, k = k_2, k = k_3\\
&T_2(n) = B_1 \cdot 2^n + B_2 \cdot k_2^n + B_3 \cdot k_3^n\\
&\log(T_2) = \theta(n)
\end{aligned}$$