Алгоритмы. ДЗ на неделю 2.

Прохоров Юрий, 771

Задача 5 (с семинара)

Вычислим $3^{11} \bmod 107$, используя алгоритм бинарного возведения в степень.

$$3^{11} = 3^{10} \cdot 3 = \left(3^5\right)^2 \cdot 3 = \left(\left(3^2\right)^2 \cdot 3\right)^2 \cdot 3$$

$$3^{11} \equiv \left(\left(3^2\right)^2 \cdot 3\right)^2 \cdot 3 \equiv (243)^2 \cdot 3 \equiv (29)^2 \cdot 3 \equiv 841 \cdot 3 \equiv 92 \cdot 3 \equiv 62 \pmod{107}$$

Сравнения по модулю верны вследствие соответствующих свойств операции взятия остатка по модулю.

Задача 7 (с семинара)

Корректность

Кратко работу алгоритма можно представить в виде:

$$multiply(x, y) = \begin{cases} 2 \cdot multiply\left(x, \left\lfloor \frac{y}{2} \right\rfloor\right), & y = 2k \neq 0, \\ x + 2 \cdot multiply\left(x, \left\lfloor \frac{y}{2} \right\rfloor\right), & y = 2k+1, \\ 0, & y = 0. \end{cases} \tag{1}$$

Докажем корректность работы алгоритма по индукции.

1. Базовый случай: $y = 0$**.** Тогда $multiply(x, 0) = 0$, что является корректным результатом.

2. Предположение индукции. Пусть на некотором шаге выполняется формула (1).

3. Общий случай. Рассмотрим два случая.

(a) Пусть y – четное. Тогда

$$multiply(x, y) = 2 \cdot multiply\left(x, \left\lfloor \frac{y}{2} \right\rfloor\right) = 2 \cdot multiply\left(x, \frac{y}{2}\right).$$

Из предположения индукции следует, что

$$multiply(x, y) = 2 \cdot multiply(x, \frac{y}{2}) = 2 \cdot x\frac{y}{2} = xy,$$

что является верным результатом.

(b) Пусть y – нечетное. Тогда

$$multiply(x, y) = x + 2 \cdot multiply\left(x, \left\lfloor \frac{y}{2} \right\rfloor\right) = x + 2 \cdot multiply\left(x, \frac{y-1}{2}\right) = x + 2x\frac{y-1}{2} = xy,$$

что также является верным результатом.

Оценка сверху по времени

На каждом шаге алгоритма число y превращается в $\left\lfloor \frac{y}{2} \right\rfloor$, что равносильно отбрасыванию последнего бита числа y. Таким образом, алгоритм выполняет $\lceil \log_2 y \rceil = O(\lceil \log_2(x+y) \rceil) = O(n)$ рекурсивных вызовов, где n – длина входа.

Если при вызове функции y оказалось четным, то происходит операция умножения на 2, а если нечетным – то, кроме умножения, производится одно сложение. Обе эти операции выполняются за $O(n)$, поэтому каждый шаг рекурсии требует время $O(n)$.

Таким образом, время работы алгоритма есть $O(n^2)$

Задача 1 (ДЗ)

При оценке функции будем пренебрегать округлениями.

Верхняя оценка:

$$f(n) = \sum_{i=1}^{n} \sqrt{i^3 + 2i + 5} \leq \sum_{i=1}^{n} \sqrt{n^3 + 2n + 5} = n\sqrt{n^3 + 2n + 5} = n^{\frac{5}{2}} \sqrt{1 + \frac{2}{n^2} + \frac{5}{n^3}} \leq \sqrt{2} n^{\frac{5}{2}}$$

при $n \geq n_2 = 2 \Longrightarrow f(n) = O(n^{\frac{5}{2}})$.

Нижняя оценка:

$$f(n) = \sum_{i=1}^{n} \sqrt{i^3 + 2i + 5} \geq \sum_{i=\frac{n}{2}}^{n} \sqrt{i^3 + 2i + 5} \geq \sum_{i=\frac{n}{2}}^{n} \sqrt{\frac{n^3}{8} + n + 5} \geq \frac{n}{2} \cdot \frac{n^{\frac{3}{2}}}{2\sqrt{2}} \sqrt{1 + \frac{1}{n^2} + \frac{40}{n^3}} \geq \frac{n^{\frac{5}{2}}}{4\sqrt{2}}$$

при $n \geq n_1 = 1 \Longrightarrow f(n) = \Omega(n^{\frac{5}{2}})$. Отсюда следует, что

$$f(n) = \Theta(n^{\frac{5}{2}}).$$

Задача 2 (ДЗ)

Пусть $g(n) = o(1)$, тогда

$$\forall \varepsilon > 0 \ \ \exists n_1 = n_1(\varepsilon) \in \mathbb{N} : \forall n \geq n_1 \longmapsto |g(n)| < \varepsilon.$$

Пусть $\varepsilon = \frac{1}{2}$, тогда найдется соответствующее число n_1. Пусть $h(n) = \Theta(n^{100})$, тогда

$$\exists C_1, C_2 > 0, n_2 \in \mathbb{N} : \forall n \geq n_2 \longmapsto C_1 n^{100} < h(n) < C_2 n^{100}.$$

Верхняя оценка:

$$\log f(n) = \log\left((3 + o(1))^n + \Theta(n^{100})\right) < \log\left((3+1)^n + C_2 n^{100}\right) = \log\left(4^n\left(1 + \frac{C_2 n^{100}}{4^n}\right)\right)$$

при $n \geq \max(n_1, n_2)$. Заметим, что

$$\frac{C_2 n^{100}}{4^n} = o(1) \Longrightarrow \exists n_3 \in \mathbb{N} : \forall n \geq n_3 \longmapsto \left|\frac{C_2 n^{100}}{4^n}\right| < 1/2$$

Пусть $n_0 = \max(n_1, n_2, n_3)$. Тогда при $n \geq n_0$:

$$\log f(n) < \log\left(2 \cdot 4^n\right) = \log 2 + n \log 4 \Longrightarrow f(n) = O(n).$$

Нижняя оценка:

$$f(n) > \log\left((3-1)^n + C_1 n^{100}\right) > \log\left(2^n\right) = n \log 2 \Longrightarrow f(n) = \Omega(n)$$

Отсюда следует, что

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

Задача 3 (ДЗ)

Для краткости обозначим переменную $bound = b$.

for *b = 0; b^2 < n; b += 1* **do**
  for *i = 0; i < b; i += 1* **do**
    for *j = 0; j < i; j += 2* **do**
      Output: алгоритм
    end
    for *j = 1; j < n; j *= 2* **do**
      Output: алгоритм
    end
  end
end

Число напечатанных слов *"алгоритм"*:

$$g(n) = \sum_{b=0}^{\lfloor\sqrt{n}\rfloor} \sum_{i=0}^{b-1} \left(\sum_{j=0}^{\lceil \frac{i}{2} \rceil - 1} 1 + \sum_{j=1}^{\lceil \log_2 n \rceil - 1} 1 \right)$$

Верхняя оценка:

$$g(n) \le \sum_{b=0}^{\lfloor\sqrt{n}\rfloor} \sum_{i=0}^{b-1} \left(\frac{i}{2} + \log_2 n \right) = \sum_{b=0}^{\lfloor\sqrt{n}\rfloor} \left(\frac{1}{2} \sum_{i=0}^{b-1} i + \log_2 n \sum_{i=0}^{b-1} 1 \right) = \sum_{b=0}^{\lfloor\sqrt{n}\rfloor} \left(\frac{1}{2} \cdot \frac{b(b-1)}{2} + \log_2 n \cdot b \right) =$$

$$= \frac{1}{4} \sum_{b=0}^{\lfloor\sqrt{n}\rfloor} b^2 - \frac{1}{4} \sum_{b=0}^{\lfloor\sqrt{n}\rfloor} b + \log_2 n \sum_{b=0}^{\lfloor\sqrt{n}\rfloor} b \le \frac{1}{24} \sqrt{n}(\sqrt{n}+1)(2\sqrt{n}+1) + (\log_2 n - \frac{1}{8})\sqrt{n}(\sqrt{n}+1) \le$$

$$\le \frac{1}{24}(2\sqrt{n})^3 + \log_2 n (2\sqrt{n})^2 = \frac{1}{3} n^{\frac{3}{2}} + 4n \log_2 n$$

при достаточно больших n. Следовательно,

$$g(n) = O(n^{3/2}).$$

Нижняя оценка:

$$g(n) \ge \sum_{b=0}^{\lfloor\sqrt{n}\rfloor} \sum_{i=0}^{b-1} \left(\frac{i}{2} + \log_2 n - 2 \right) = \sum_{b=0}^{\lfloor\sqrt{n}\rfloor} \left(\frac{1}{2} \sum_{i=0}^{b-1} i + (\log_2 n - 2) \sum_{i=0}^{b-1} 1 \right) = \sum_{b=0}^{\lfloor\sqrt{n}\rfloor} \left(\frac{1}{2} \cdot \frac{b(b-1)}{2} + (\log_2 n - 2) \cdot b \right) =$$

$$= \frac{1}{4} \sum_{b=0}^{\lfloor\sqrt{n}\rfloor} b^2 - \frac{1}{4} \sum_{b=0}^{\lfloor\sqrt{n}\rfloor} b + (\log_2 n - 2) \sum_{b=0}^{\lfloor\sqrt{n}\rfloor} b \ge \frac{1}{24}(\sqrt{n}-1)\sqrt{n}(2\sqrt{n}-1) + (\log_2 n - 2 - \frac{1}{8})(\sqrt{n}-1)\sqrt{n} \ge$$

$$\ge \frac{1}{24}(\sqrt{n})^3 + (\log_2 n - 3)(\frac{1}{2}\sqrt{n})^2 = \frac{1}{24} n^{\frac{3}{2}} + \frac{n}{4} \log_2 n - \frac{3n}{4}$$

при достаточно больших n. Следовательно,

$$g(n) = \Omega(n^{3/2}).$$

Из этих оценок следует, что

$$g(n) = \Theta(n^{3/2}).$$

Задача 4 (ДЗ)

а)

$$238x + 385y = 133$$

Используем расширенный алгоритм Евклида:

$$238\begin{pmatrix}1\\0\end{pmatrix} \qquad 385\begin{pmatrix}0\\1\end{pmatrix}$$

$$385\begin{pmatrix}0\\1\end{pmatrix} \qquad 238\begin{pmatrix}1\\0\end{pmatrix}$$

$$238\begin{pmatrix}1\\0\end{pmatrix} \qquad 147\begin{pmatrix}-1\\1\end{pmatrix}$$

$$147\begin{pmatrix}-1\\1\end{pmatrix} \qquad 91\begin{pmatrix}2\\-1\end{pmatrix}$$

$$91\begin{pmatrix}2\\-1\end{pmatrix} \qquad 56\begin{pmatrix}-3\\2\end{pmatrix}$$

$$56\begin{pmatrix}-3\\2\end{pmatrix} \qquad 35\begin{pmatrix}5\\-3\end{pmatrix}$$

$$35\begin{pmatrix}5\\-3\end{pmatrix} \qquad 21\begin{pmatrix}-8\\5\end{pmatrix}$$

$$21\begin{pmatrix}-8\\5\end{pmatrix} \qquad 14\begin{pmatrix}13\\-8\end{pmatrix}$$

$$14\begin{pmatrix}13\\-8\end{pmatrix} \qquad 7\begin{pmatrix}-21\\13\end{pmatrix}$$

Левый аргумент кратен правому, поэтому $\gcd(238, 385) = 7$. Коэффициенты при 7 есть частное решение уравнения

$$238x + 385y = \gcd(238, 385) = 7$$

Следовательно, частное решение уравнения $238x + 385y = 133 = 7n, n = 19$:

$$\begin{pmatrix}x_0\\y_0\end{pmatrix} = n\begin{pmatrix}-21\\13\end{pmatrix} = \begin{pmatrix}-399\\247\end{pmatrix}$$

Общее решение получим по формуле

$$\begin{pmatrix}x\\y\end{pmatrix} = \begin{pmatrix}x_0\\y_0\end{pmatrix} + k_1\begin{pmatrix}\frac{385}{\gcd(238,385)}\\-\frac{238}{\gcd(238,385)}\end{pmatrix}$$

$$\begin{pmatrix}x\\y\end{pmatrix} = \begin{pmatrix}-399\\247\end{pmatrix} + k_1\begin{pmatrix}55\\-34\end{pmatrix}$$

$$\begin{pmatrix}x\\y\end{pmatrix} = \begin{pmatrix}41\\-25\end{pmatrix} + k\begin{pmatrix}55\\-34\end{pmatrix}$$

б)

$$143x + 121y = 52$$

Используем расширенный алгоритм Евклида:

$$143\begin{pmatrix}1\\0\end{pmatrix} \qquad 121\begin{pmatrix}0\\1\end{pmatrix}$$

$$121\begin{pmatrix}0\\1\end{pmatrix} \qquad 22\begin{pmatrix}1\\-1\end{pmatrix}$$

$$22\begin{pmatrix}1\\-1\end{pmatrix} \qquad 11\begin{pmatrix}-5\\6\end{pmatrix}$$

$gcd(143, 121) = 11,\ 52 \neq 11n \implies$ уравнение не имеет целочисленных решений.

Задача 5 (ДЗ)

Корректность

Алгоритм возвращает два значения: целую часть от деления и остаток. Обозначим их следующим образом:

$$Divide(x, y) = (Div(x, y), Mod(x, y))$$

Работу алгоритма можно представить в следующем развернутом виде:

$$(q, p) = \begin{cases} 0, & x = 0, \\ (Div(\lfloor \frac{x}{2} \rfloor, y), Mod(\lfloor \frac{x}{2} \rfloor, y)), & x > 0; \end{cases}$$

$$r = \begin{cases} 2p, & x = 2k, \\ 2p + 1, & x = 2k + 1; \end{cases}$$

$$Mod(x, y) = \begin{cases} r, & r < y, \\ r - y, & r \geq y; \end{cases}$$

$$Div(x, y) = \begin{cases} 2q, & r < y, \\ 2q + 1, & r \geq y; \end{cases}$$

Докажем корректность работы алгоритма по индукции. Базовый случай $x = 0$ разрешается верно. Предположим, что на некотором ненулевом входе алгоритм выдает корректное значение. Покажем, что на входе (x, y) тоже будет выведено верное значение.

1. x **– четное.** Тогда $x = 2t$. По предположению индукции:

$$Divide(\left\lfloor \frac{x}{2} \right\rfloor, y) = Divide(t, y) = (q, p).$$

Следовательно, выполняются следующие условия:

$$t = qy + p, \qquad 0 \leq q < y \Longrightarrow$$
$$\Longrightarrow x = 2t = 2qy + 2p = 2qy + r, \qquad 0 \leq r < 2y$$

Пусть оказалось $r < y$, тогда

$$x = 2qy + r = Div(\left\lfloor \frac{x}{2} \right\rfloor, y) \cdot y + Mod(\left\lfloor \frac{x}{2} \right\rfloor, y),$$

что удовлетворяет математическому смыслу деления с остатком.
Пусть оказалось $r \geq y$. Из условия $r < 2y$ следует, что $r - y < y$. Поэтому

$$x = 2qy + r = 2qy + y + r - y = (2q + 1)y + r - y = Div(\left\lfloor \frac{x}{2} \right\rfloor, y) \cdot y + Mod(\left\lfloor \frac{x}{2} \right\rfloor, y),$$

что верно.

2. x **– нечетное.** Тогда $x = 2t + 1$. По предположению индукции:

$$Divide(\left\lfloor \frac{x}{2} \right\rfloor, y) = Divide(t, y) = (q, p).$$

Следовательно, выполняются следующие условия:

$$t = qy + p, \qquad 0 \leq q < y \Longrightarrow$$
$$\Longrightarrow x = 2t + 1 = 2qy + 2p + 1 = 2qy + r, \qquad 0 \leq r < 2y.$$

Далее аналогично предыдущему случаю.

Оценка сверху по времени

При каждом следующем вызове функции от числа x отбрасывается последний бит. Поэтому всего будет $O(n)$ рекурсивных вызовов. При каждом вызове выполняется конечное число операций сложения, вычитания, сравнения, умножения на 2, каждая из которых занимает $O(n)$ времени, т.е. в сумме они тоже занимают $O(n)$ времени. Поэтому весь алгоритм работает за время $O(n^2)$.