1 Глава 1

1.1 Поиск контрпримеров

1.1.1 #1

Условие. Докажите, что значение $a+b$ может быть меньшим, чем значение $\min(a, b)$.

Допустим:

$$a = 0, b = -1 \tag{1}$$

Тогда:

$$\begin{align} 0 + (-1) = & \quad -1 \tag{2} \\ & \quad -1 \quad < 0 = \min(0, 1) \tag{3} \end{align}$$

Что и требовалось доказать.

1.1.2 #2

Условие. Докажите, что значение $a \times b$ может быть меньшим, чем значение $\min(a, b)$.

Допустим:

$$a = \frac{1}{2}, b = \frac{1}{4} \tag{4}$$

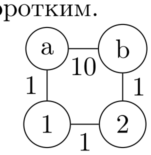
Тогда:

$$\begin{align} \frac{1}{2} \times \frac{1}{4} = \frac{1}{8} \tag{5} \\ \tfrac{1}{8} < \min(\tfrac{1}{2}, \tfrac{1}{4}) = \tfrac{1}{4} \tag{6} \end{align}$$

Что и требовалось доказать.

1.1.3 #3

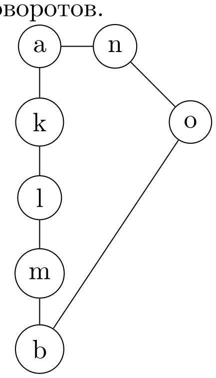
Условие. Начертите сеть дорог с двумя точками a и b, такими, что маршрут между ними, преодолеваемый за кратчайшее время, не является самым коротким.



Цена маршрута «a,1,2,b» равна 3, а цена маршрута «a,b» равна 10. То есть, маршрут, проходящий через больше точек, более длинный, преодолевается за меньшее время. Что и требовалось доказать.

1.1.4 #4

Условие. Начертите сеть дорог с двумя точками a и b, самый короткий маршрут между которыми не является маршрутом с наименьшим числом поворотов.



В данной сети дорог кратчайший маршрут от a до b это $anob$, в котором 3 ребра, но два поворота. Однако, маршрут без поворотов вообще, $aklmb$ не самый короткий — в нём 4 ребра. Что и требовалось доказать.

1.1.5 #5

Условие. Задача о рюкзаке: имея множество целых чисел $S = \{s_1, s_2, \ldots, s_n\}$ и целевое число T, найти такое подмножество множества S, сумма которого в точности равна T. Например, множество $S = \{1, 2, 5, 9, 10\}$, содержит такое подмножество, сумма элементов которого равна T = 22, но не T = 23.

Найти контрпримеры для каждого из следующих алгоритмов решения задачи о рюкзаке, т. е., нужно найти такое множество S и число T, при которых подмножество, выбранное с помощью данного алгоритма, не до конца заполняет рюкзак, хотя правильное решение и существует:

- вкладывать элементы множества S в рюкзак в порядке слева направо, если они подходят (т. е., алгоритм «первый подходящий»);
- вкладывать элементы множества S в рюкзак в порядке от наименьшего до наибольшего (т. е., используя алгоритм «первый лучший»);
- вкладывать элементы множества S в рюкзак в порядке от наибольшего до наименьшего.

Первый подходящий. $T = 10, S = \{9, 2, 2, 2, 2, 2\}$
Первый лучший. $T = 6, S = \{1, 2, 5, 4\}$
От самого большого. $T = 6, S = \{3, 2, 5, 4\}$

1.1.6 #6

Условие. Задача о покрытии множества: имея семейство подмножеств $S_1, \ldots, S_m$ универсального множества $U = \{1, \ldots, n\}$, найдите семейство подмножеств

$T \subset S$ наименьшей мощности, чтобы $\bigcup_{t_i} \in T^{t_i} = U$. Например, для семейства подмножеств $S_1 = \{1, 3, 5\}$, $S_2 = \{2, 4\}$, $S_3 = \{1, 4\}$, $S_5 = \{2, 5\}$ покрытием множества будет семейство подмножеств S_1 и S_2.

Приведите контрпример для следующего алгоритма: выбираем самое мощное подмножество для покрытия, после чего удаляем все его элементы из универсального множества; повторяем добавление подмножества, содержащего наибольшее количество неохваченных элементов, пока все элементы не будут покрыты.

$$S_1 = \{1, 3, 5, 7, 9\} \tag{7}$$
$$S_2 = \{1, 2\} \tag{8}$$
$$S_3 = \{3, 4\} \tag{9}$$
$$S_4 = \{5, 6\} \tag{10}$$
$$S_5 = \{7, 8\} \tag{11}$$
$$S_6 = \{9, 10\} \tag{12}$$

Наилучшее покрытие это $S_2 \cup S_3 \cup S_4 \cup S_5 \cup S_6$, но жадный алгоритм вначале выберет S_1 как самое мощное подмножество, а потом не останется ничего другого, как добавлять все остальные подмножества, то есть, в итоге алгоритм выберет все исходные подмножества в качестве результата работы, $T = S$.

1.2 Доказательство правильности

1.2.1 #7

Условие. Докажите правильность следующего рекурсивного алгоритма умножения двух натуральных чисел для всех целочисленных констант $c \geqslant 2$:

function MULTIPLY(y, z)
  if $z = 0$ **then**
    return 0
  else
    return $multiply(cy, \lfloor z/c \rfloor) + y * (z \mod c)$
  end if
end function

Вычислим результат работы multiply на каком-нибудь простом примере:

$$
\begin{aligned}
& c = 2, y = 2, z = 3 && (13)\\
& multiply(2, 3) && (14)\\
= \quad & multiply(2 * 2, \lfloor 3/2 \rfloor) + 2 * (3 \mod 2) && (15)\\
= \quad & multiply(4, 1) + 2 * 1 && (16)\\
= \quad & multiply(4, 1) + 2 && (17)\\
= \quad & multiply(4 * 2, \lfloor 1/2 \rfloor) + 4 * (1 \mod 2) + 2 && (18)\\
= \quad & multiply(8, 0) + 4 * 1 + 2 && (19)\\
= \quad & 0 + 4 + 2 && (20)\\
= \quad & 6 && (21)
\end{aligned}
$$

Пусть $z = 0$. Тогда $multiply(y, z) = 0$.

Допустим теперь, что алгоритм правильный, то есть $multiply(y, z) = yz$ при:

$$z \leqslant n \tag{22}$$

$$y \geqslant 1 \tag{23}$$

$$c \geqslant 2 \tag{24}$$

Теперь докажем, что $multiply(y, n + 1) = y(n + 1)$.

$$mutiply(y, n + 1) = \tag{25}$$

$$multiply(y * c, \lfloor \frac{n+1}{c} \rfloor) + y * ((n + 1) \mod c) = \tag{26}$$

$$y * c * \lfloor \frac{n+1}{c} \rfloor + y * ((n + 1) \mod c) = \tag{27}$$

$$y * (c * \lfloor \frac{n+1}{c} \rfloor + (n + 1) \mod c) \tag{28}$$

Теперь докажем, что $c * \lfloor \frac{n+1}{c} \rfloor + (n + 1) \mod c) = n + 1$.

$$n + 1 \equiv z \tag{29}$$

$$c\lfloor \frac{z}{c} \rfloor + z \mod c = z \tag{30}$$

z/c в случае деления произвольных натуральных чисел — операция с остатком. $\lfloor z/c \rfloor$ же избавляется от этого остатка.

В то же время, $z \mod c$ это операция получения *остатка от деления* z на c.

Формула (30) может использоваться как есть, потому что следует из определения операций, которые в ней используются.

Если мы умножим c на операцию, убирающую остаток от деления z на c, то получим как бы «восстановленную» z без остатка после деления.

$$3 \times \lfloor \frac{10}{3} \rfloor = 3 \times 3 = 9 \tag{31}$$

Но если мы потом прибавим буквально остаток от этого же деления, мы получим исходное число!

$$10 \mod 3 = 1 \tag{32}$$

А $9 + 1 = 10$, как и следует из объяснения выше.

Таким образом, из (30) и (28) следует:

$$y \times (c \times \lfloor \frac{n+1}{c} \rfloor + (n+1) \mod c) = y \times (n+1) \tag{33}$$

$$\Rightarrow multiply(y, n+1) = y \times (n+1) \tag{34}$$

Что и требовалось доказать.

1.2.2 #8

Условие. Докажите правильность следующего алгоритма вычисления полинома $P(x) = a_n x^n + a_{n-1} x^{n-1} + \ldots + a_1 x^{+} a_0$:

```
function HORNER(A, x)
    p = A_n
    for i ← n − 1, 0 do
        p = p × x + A_i
    end for
    return p
end function
```

Пусть $n = 0$. Тогда цикл не выполнится ни разу и результатом будет A_0.

Пусть $n = 1$. Тогда:

$$p = A_1 \tag{35}$$

$$i = 0: \quad p = A_1 \times x + A_0 \tag{36}$$

Пусть $n = 2$. Тогда:

$$p = A_2 \tag{37}$$

$$i = 1: \quad p = A_2 \times x + A_1 \tag{38}$$

$$i = 0: \quad p = (A_2 \times x + A_1) \times x + A_0 = A_2 \times x^2 + A_1 \times x + A_0 \tag{39}$$

Допустим, что алгоритм правильный при любых n. Посмотрим, что происходит при $n + 1$:

$$
\begin{aligned}
& p = A_{n+1} && (40)\\
i = n: \quad & p = A_{n+1} \times x + A_n && (41)\\
i = n-1: \quad & p = (A_{n+1} \times x + A_n) \times x + A_{n-1} && (42)\\
\dots \quad & && (43)\\
i = n-n = 0: \quad & p = A_{n+1}x^{n+1} + A_n x^n + A_{n-1}x^{n-1} + \ldots + A_{n-n} && (44)
\end{aligned}
$$

Что и требовалось доказать.

1.2.3 #9

Условие. Докажите правильность следующего алгоритма сортировки:

```
function bubblesort(A: list[1...n])
    var int i,j
    for i from n to 1
        for j from 1 to i - 1
            if (A[j] > A[j+1])
                swap A[j] and A[j + 1]
```

Пусть $n = 2$. Тогда:

$$i = 2, j = 1: \quad A[1] > A[2]?swap : keep \tag{45}$$

Пусть алгоритм может отсортировать любой массив длиной n. Может ли он отсортировать массив длиной $n+1$?

Передача в этот алгоритм массив длиной $n+1$ эквивалентна замене его на следующий код:

```
function bubblesort(A: list[1...n+1])
    var int i,j
    for j from 1 to n + 1 - 1
        if (A[j] > A[j + 1])
            swap A[j] and A[j + 1]
    for i from n to 1
        for j from 1 to i - 1
            if (A[j] > A[j+1])
                swap A[j] and A[j + 1]
```

По индукции мы допустили, что основная пара вложенных циклов действительно отсортирует все элементы от 1 до n.

Мы же добавили в код ещё один цикл:

```
for j from 1 to n
    if (A[j] > A[j + 1])
        swap A[j] and A[j + 1]
```

Этот цикл проверит два новых элемента: `A[n]` и `A[n+1]`, последние в списке. Элемент `A[n+1]` идёт после отсортированных элементов `A[1] ... A[n]`. Если они не в правильном порядке, он их поменяет местами, в точности как случай когда $n = 2$.

Таким образом, этот алгоритм действительно сортирует массивы любой длины.

1.3 Математическая индукция

1.3.1 #10

Условие. Докажите, что:

$$\sum_{i=1}^{n} i = \frac{n(n+1)}{2} \tag{46}$$

для $n \geqslant 0$.

По индукции, пусть $n = 1$:

$$1 = \frac{1 \times (1+1)}{2} \tag{47}$$

$$1 = \frac{1 \times 2}{2} \tag{48}$$

$$1 = 1 \tag{49}$$

Пусть верно для n. Вычислим для $n + 1$:

$$1 + 2 + \ldots + n + (n+1) = \frac{(n+1)(n+2)}{2} \tag{50}$$

$$\frac{n(n+1)}{2} + n + 1 = \frac{(n+1)(n+2)}{2} \tag{51}$$

$$n(n+1) + 2n + 2 = (n+1)(n+2) \tag{52}$$

$$n^2 + n + 2n + 2 = n^2 + n + 2n + 2 \tag{53}$$

Что и требовалось доказать.

1.3.2 #11

Условие. Докажите, что:

$$\sum_{i=1}^{n} i^2 = n(n+1)(2n+1)/6, n \geqslant 0 \tag{54}$$

Пусть $n = 1$, тогда:

$$1^2 = \frac{1 \times (1+1)(2 \times 1 + 1)}{6} \tag{55}$$
$$1 = \frac{6}{6} \tag{56}$$
$$1 = 1 \tag{57}$$

Теперь пусть формула верна для любых n. Вычислим для $n+1$:

$$\sum_{i=1}^{n} i^2 + (n+1)^2 = \frac{(n+1)(n+1+1)(2 \times (n+1) + 1)}{6} \tag{58}$$
$$\frac{n(n+1)(2n+1)}{6} + (n+1)^2 = \frac{(n+1)(n+2)(2n+3)}{6} \tag{59}$$
$$n(n+1)(2n+1) + 6n^2 + 12n + 6 = (n+1)(n+2)(2n+3) \tag{60}$$
$$(n^2+n)(2n+1) + 6n^2 + 12n + 6 = (n^2+n+2n+2)(2n+3) \tag{61}$$
$$2n^3 + 2n^2 + n^2 + n + 6n^2 + 12n + 6 = 2n^3 + 2n^2 + 4n^2 + 4n + 3n^2 + 3n + 6n + 6 \tag{62}$$
$$2n^3 + 9n^2 + 13n + 6 = 2n^3 + 9n^2 + 13n + 6 \tag{63}$$

Что и требовалось доказать.

1.3.3 #12

Условие. Докажите, что:

$$\sum_{i=1}^{n} i^3 = \frac{n^2(n+1)^2}{4}, n \geqslant 0 \tag{64}$$

Пусть $n = 1$.

$$1 = \frac{1 \times 2^2}{4} \tag{65}$$
$$1 = 4/4 \tag{66}$$
$$1 = 1 \tag{67}$$

Теперь пусть формула верна для любых n. Вычислим для $n+1$:

$$\sum_{i=1}^{n} i^3 + (n+1)^3 = \frac{(n+1)^2(n+2)^2}{4} \tag{68}$$
$$\frac{n^2(n+1)^2}{4} + (n+1)^3 = \frac{(n+1)^2(n+2)^2}{4} \tag{69}$$
$$n^2(n+1)^2 + 4(n+1)(n+1)^2 = (n+1)^2(n+2)^2 \tag{70}$$
$$(n+1)^2(n^2+4n+4) = (n+1)^2(n+2)^2 \tag{71}$$
$$n^2 + 4n + 4 = (n+2)^2 \tag{72}$$
$$n^2 + 4n + 4 = n^2 + 4n + 4 \tag{73}$$

Что и требовалось доказать.

1.3.4 #13

Условие. Докажите, что:

$$\sum_{i=1}^{n} i(i+1)(i+2) = \frac{n(n+1)(n+2)(n+3)}{4}, n \geqslant 0 \tag{74}$$

Пусть $n = 1$.

$$2 \times 3 = \frac{2 \times 3 \times 4}{4} \tag{75}$$

$$6 = 6 \tag{76}$$

Теперь пусть формула верна для любых n. Вычислим для $n+1$:

$$\sum_{i=1}^{n} i(i+1)(i+2) + (n+1)(n+2)(n+3) = \frac{(n+1)(n+2)(n+3)(n+4)}{4} \tag{77}$$

$$\frac{n(n+1)(n+2)(n+3)}{4} + (n+1)(n+2)(n+3) = \frac{(n+1)(n+2)(n+3)(n+4)}{4} \tag{78}$$

$$n(n+1)(n+2)(n+3) + 4(n+1)(n+2)(n+3) = (n+1)(n+2)(n+3)(n+4) \tag{79}$$

$$n+4 = n+4 \tag{80}$$

Что и требовалось доказать.

1.3.5 #14

Условие. Докажите, что:

$$\sum_{i=0}^{n} a^i = \frac{a^{n+1}-1}{a-1}, n \geqslant 1, a \neq 1 \tag{81}$$

Пусть $n = 1$. Тогда:

$$a^0 + a^1 = \frac{a^2-1}{a-1} \tag{82}$$

$$(a-1)(a+1) = a^2 - 1 \tag{83}$$

$$a^2 - a + a - 1 = a^2 - 1 \tag{84}$$

$$a^2 - 1 = a^2 - 1 \tag{85}$$

Теперь пусть формула верна для любых n. Вычислим для $n+1$:

$$\sum_{i=0}^{n} a^i + a^{n+1} = \frac{a^{n+2}-1}{a-1} \tag{86}$$
$$\frac{a^{n+1}-1}{a-1} + a^{n+1} = \frac{a^{n+2}-1}{a-1} \tag{87}$$
$$a^{n+1} - 1 + a^{n+1}(a-1) = a^{n+2} - 1 \tag{88}$$
$$a^{n+1} - 1 + a^{n+2} - a^{n+1} = a^{n+2} - 1 \tag{89}$$
$$a^{n+2} - 1 = a^{n+2} - 1 \tag{90}$$

Что и требовалось доказать.

1.3.6 #15

Условие. Докажите, что:

$$\sum_{i=1}^{n} \frac{1}{i(i+1)} = \frac{n}{n+1}, n \geqslant 0 \tag{91}$$

Пусть $n = 1$. Тогда:

$$1/2 = 1/2 \tag{92}$$

Теперь пусть формула верна для любых n. Вычислим для $n + 1$:

$$\sum_{i=1}^{n} \frac{1}{i(i+1)} + \frac{1}{(n+1)(n+2)} = \frac{n+1}{n+2} \tag{93}$$
$$\frac{n}{n+1} + \frac{1}{(n+1)(n+2)} = \frac{n+1}{n+2} \tag{94}$$
$$\frac{n(n+2)}{(n+1)(n+2)} + \frac{1}{(n+1)(n+2)} = \frac{(n+1)(n+1)}{(n+1)(n+2)} \tag{95}$$
$$\frac{n(n+2)+1}{(n+1)(n+2)} = \frac{n^2+2n+1}{(n+1)(n+2)} \tag{96}$$
$$\frac{n^2+2n+1}{(n+1)(n+2)} = \frac{n^2+2n+1}{(n+1)(n+2)} \tag{97}$$

Что и требовалось доказать.

1.3.7 #16

Условие. Докажите, что $n^3 + 2n$ делится на 3 для $n \geqslant 0$.

Пусть $n = 0$. Тогда выражение равно 0, что условно делится на 3.

Пусть выражение верно для любых n.

Вычислим для $n+1$.

$$(n+1)^3 + 2(n+1) = \tag{98}$$
$$(n+1)(n^2+2n+1) + 2n + 2 = \tag{99}$$
$$n^3 + 2n^2 + n + n^2 + 2n + 1 + 2n + 2 = \tag{100}$$
$$n^3 + 3n^2 + 5n + 3 = \tag{101}$$
$$n^3 + 2n + 3n^2 + 3n + 3 = \tag{102}$$
$$n^3 + 2n + 3(n^2 + n + 1) \tag{103}$$

Представим (103) в виде $A+B$, где $A = n^3+2n$, а $B = 3(n^2+n+1)$. Мы допустили по индукции, что $A \vdots 3$. В то же время B является произведением 3 на натуральное число больше 0, то есть, тоже делится на 3. По свойству делимости сумма чисел, которые делятся на 3, делится на 3.

Что и требовалось доказать.

1.3.8 #17

Условие. Докажите, что дерево с n вершинами имеет в точности $n-1$ рёбер.

Если $n = 0$ то задача не имеет смысла. Если $n = 1$ то дерево с одной вершиной не имеет рёбер, то есть, верно. Если $n = 2$ то между двумя вершинами можно провести только одно ребро, то есть, верно.

Пусть верно для всех n.

Дерево с $n+1$ вершинами это дерево с n вершинами, к которому присоединили ещё одну. Дерево с n вершинами имеет $n-1$ рёбер по индукции.

Так как у нас не граф, а дерево, есть только один способ добавить вершину в дерево: соединив её ровно одним ребром с другой одной существующей вершиной в дереве.

Так как таким образом мы добавили одну вершину и одно ребро, соотношение между количеством рёбер и количеством вершин не изменилось, у нас всё так же ровно на одно ребро меньше, чем вершин.

Что и требовалось доказать.

1.3.9 #18

Условие. Докажите, что сумма кубов первых n положительных целых чисел равна квадрату суммы этих целых чисел, т. е.,

$$\sum_{i=1}^{n} i^3 = \left(\sum_{i=1}^{n} i\right)^2 \tag{104}$$

Ранее в задачах #10 и #12 были доказаны сокращённые формы записи этих сумм. Подставим их в выражение и упростим:

$$\frac{n^2(n+1)^2}{4} = \left(\frac{n(n+1)}{2}\right)^2 \quad (105)$$

$$\frac{n^2(n+1)^2}{4} = \frac{n^2(n+1)^2}{4} \quad (106)$$

Что и требовалось доказать.

1.4 Приблизительные подсчёты

1.4.1 #19

Условие. Содержат ли все ваши книги, по крайней мере, миллион страниц? Каково общее количество всех книг в вашей институтской библиотеке?