1. Рекуррентные соотношения:

a.

A[n] = A[n-1] + 1

A[1] = 3

A[n] – A[n-1] = 1

A[n-1] – A[n-2] = 1

…

A[2] – A[1] = 1

Значит, просуммировав все эти выражения, мы получим:

A[n] – A[1] = n-1 => A[n] = A[1] + n-1 = 3 + n-1 = n+2

A[1] = 1+2 = 3 – верно.

A[n] = (n-1)+2+1 = n+2. Верно!

b.

A[n] = A[n-1] + n

A[0] = 2

A[n] – A[n-1] = n

A[n-1] – A[n-2] = n-1

…

A[2] – A[1] = 2

A[1] – A[0] = 1

Значит, просуммировав все эти выражения, мы получим:

A[n] – A[0] = 1 + 2 + 3 + … + n = n\*(n+1)/2

A[n] = n\*(n+1)/2 + 2

A[0] = 0\*1/2 + 2 = 2 – верно.

A[n] = (n-1)\*n/2 + 2 + n = (n+1)\*n/2 + 2. Верно!

c.

A[n] = 2\*A[n-1] + 2

A[0] = 1

A[n] – 2\*A[n-1] = 2

2\*A[n-1] – 4\*A[n-2] = 4

…

2^(n-2)\*A[2] – 2^(n-1)\*A[1] = 2^(n-1)

2^(n-1)\*A[1] – 2^n\*A[0] = 2^n

Значит, просуммировав все эти выражения, мы получим:

A[n] – 2^n \* A[0] = 2 + 4 + 8 + … + 2^n = 2^(n+1) – 2 => A[n] = 2^n + 2^(n+1)-2 = 3\*2^n - 2

A[0] = 3\*2^1-2 = 3-2 = 1 – верно.

A[n] = 2\*A[n-1] + 2 = 2 \* (3 \* 2^(n-1)-2) + 2 = 3 \* 2^n – 2. Верно!

d.

A[n] = 4\*A[n-1] + 5\*A[n-2]

A[0] = 1

A[1] = 17

Будем искать производящую функцию G(z) = A[0] + z\*A[1] + z^2\*A[2] + …

Для этого умножим каждый из A[k] на z^k

A[0] = 1\*1

A[1] = 17\*z

…

A[k] = 4\*A[k-1]\*z^k + 5\*A[k-2]\*z^k

Теперь сложим все уравнения:

Значит

Также

Значит A[n] =

A[0] = 3 – 2 = 1 – верно.

A[1] = 3\*5 – 2\*(-1) = 15+2 = 17 – верно.

A[n] = 4\*A[n-1] + 5\*A[n-2] = 12\*5^(n-1) - 8\*(-1)^(n-1) + 15\*5^(n-2) – 10\*(-1)^(n-2) =

= (5^(n-2))\*(60+15) + ((-1)^(n-2))\*(8–10) = 5^2 \* 5^(n-2) \* 3 + (-1)^2 \* (-1)^(n-2) \* (-2) =

= 3\*5^n – 2\*(-1)^n. Верно!

e.

A[n] = 6\*A[n-1] - 9\*A[n-2]

A[0] = 2

A[1] = 3

Будем искать производящую функцию G(z) = A[0] + z\*A[1] + z^2\*A[2] + …

Для этого умножим каждый из A[k] на z^k

A[0] = 2\*1

A[1] = 3\*z

…

A[k] = 6\*A[k-1]\*z^k - 9\*A[k-2]\*z^k

Теперь сложим все уравнения:

Значит

Также

Значит A[n] =

A[0] = 2\*1 = 2 – верно.

A[1] = 1\*3 = 3 – верно.

A[n] = 6\*A[n-1] – 9\*A[n-2] = (12-6n) \* 3^(n-1) – (18-9n) \* 3^(n-2) = (4-2n)\*3^n – (2-n)\*3^n =

= (2-n)\*3^n. Верно!

f.

A[n] = 2\*A[n−1] + A[n−2] – 2\*A[n−3]

A[0] = 3

A[1] = 2

A[2] = 6

Получим характеристическое уравнение, путём подставления r^k вместо A[k] в уравнении рекурсивного перехода:

r^n = 2\*r^(n-1) + r^(n-2) – 2\*r^(n-3)

r^(n-3) \* (r^3 – 2r^2 - r + 2) = 0

r^(n-3) \* (r^2\*(r – 2) – (r – 2)) = 0

r^(n-3) \* (r^2 - 1) \*(r – 2) = 0

r^(n-3) \* (r+1)\*(r – 1) \*(r – 2)) = 0

Нас интересуют ненулевые корни этого уравнения, то есть

r = -1

r = 1

r = 2

Значит A[n] = a\*(-1)^n + b\*1^n + c\*2^n

Найдём a, b и с:

A[0] = a + b + c = 3

A[1] = -a + b + 2c = 2

A[2] = a + b + 4c = 6

Значит a = 1, b = 1, c = 1. То есть A[n] = (-1)^n + 2^n + 1

Проверка:

A[0] = 1 + 1 + 1 = 3 – верно.

A[1] = -1 + 1 + 2 = 2 – верно.

A[2] = 1 + 1 + 4 = 6 – верно.

A[n] = 2\*A[n-1] + A[n-2] – 2\*A[n-3] = 2\*(-1)^(n-1) + 2\*2^(n-1) + 2 + (-1)^(n-2) + 2^(n-2) + 1 –

- 2\*(-1)^(n-3) – 2\*2^(n-3) – 2 = (-1)^n\*(-2+1+2) + 2^n\*(1 + ½ – ½ ) + 2 + 1 – 2 =

= (-1)^n + 2^n + 1. Верно!

В пунктах a, b, c мы решали рекуррентные соотношения, путём нахождения суммы разностей между парами соседних элементов.

В пунктах d и e использовалась производящая функция последовательности, с помощью которой и получилось найти искомое выражение.

В пункте f использовался упрощённый метод предыдущего, при помощи характеристических уравнений.

1. Подсчёты асимптотики в рекурсивных формулах.
2. T(n) = 2T(n/2) + n

Используем 2a case) мастер-теоремы, т. к.

c\_crit = logb(a) = log2(2) = 1

c = 1 => c = c\_crit

k = 0 > -1

Значит итоговая асимптотика будет: θ(n^c \* log2(n)^k) = θ(n\*log2(n))

1. T(n) = T(3n/4) + T(n/4) + n

Воспользуемся Akra–Bazzi методом, для этого решим такое уравнение:

(3/4)^p + (1/4)^p = 1

Левая часть уравнения – монотонно убывающая функция, соответственно равна константе она может быть максимум при одном аргументе. Этот аргумент p = 1.

1. T(n) = 3T(n/2) + n

c\_crit = log2(3) > c = 1

Значит используем 1 case) мастер-теоремы.

Значит итоговая асимптотика будет: θ(n^c\_crit) = θ(n^log2(3))

1. T(n) = 2T(n/2) +n/log2(n)

Используем 2b case) мастер-теоремы, т. к.

c\_crit = logb(a) = log2(2) = 1

c = 1 => c = c\_crit

k = -1

Значит итоговая асимптотика будет: θ(n^c \* log2(log2(n))) = θ(n\*log2(log2(n)))

1. T(n) = 6T(n/3) + n^2 \* log2(n)

c\_crit = log3(6) < c = 2

Значит используем 3 case) мастер-теоремы.

Значит итоговая асимптотика будет: θ(f(n)) = θ(n^2 \* log2(n))

1. T(n) = T(3n/4) + n\*log2(n)

c\_crit = log4/3(1) = 0 < c = 1

Значит используем 3 case) мастер-теоремы.

Значит итоговая асимптотика будет: θ(f(n)) = θ(n \* log2(n))

1. 

(1/2)^p + (1/2)^p = 1

Левая часть уравнения – монотонно убывающая функция, соответственно равна константе она может быть максимум при одном аргументе. Этот аргумент p = 1.

1. T(n) = T(n/2) + T(n/4) + 1

(1/2)^p + (1/4)^p = 1

Сделаем замену (1/2)^p = t > 0.

t^2 + t – 1 = 0

D = 1 + 4\*1 = 5

Т. к. t > 0, то подходящий корень только

2^(-p) =

p = => 0 < p < 1 =>

1. T(n) = T(n/2) + T(n/3) + T(n/6) + n

(1/2)^p + (1/3)^p + (1/6)^p = 1

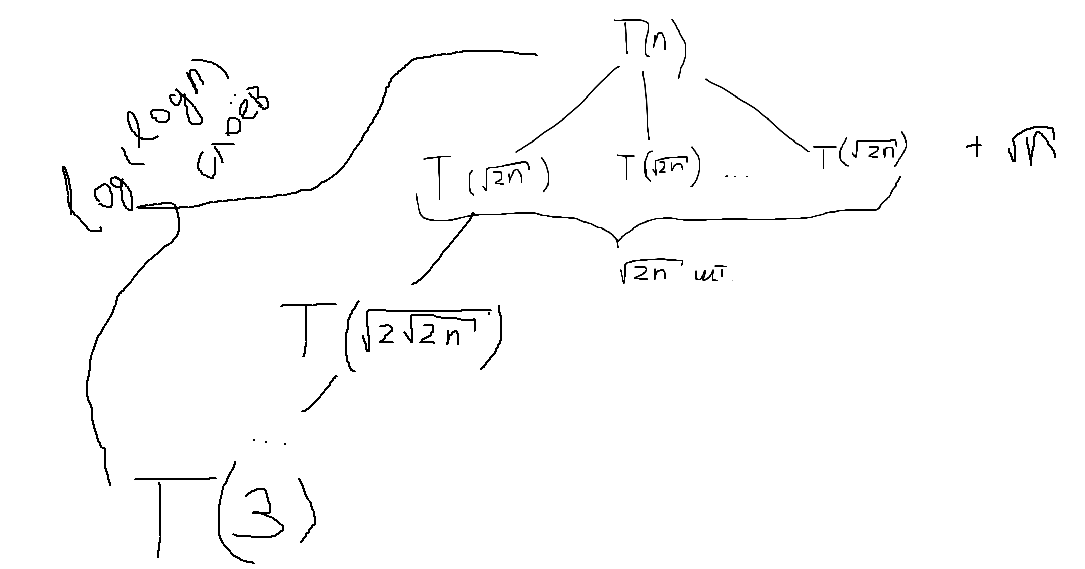
Левая часть уравнения – монотонно убывающая функция, соответственно равна константе она может быть максимум при одном аргументе. Этот аргумент p = 1.

1. T (n) = 2T(n/3) + 2T(2n/3) + n

2\*(1/3)^p + 2\*(2/3)^p = 1

p ≈ 2.196291818 > 1

1. 



Дерево рекурсии будет таким:

Возможно непонятно, почему такое кол-во слоёв?

Мы должны остановиться на 3, потому что рекурсия должна где-то останавливаться, а меньше 3 уже некуда, т. к. там будет зацикливание.

Найдём номер слоя, на котором прервётся функция:

sqrt(2sqrt(2sqrt(…(2sqrt(2n)))) = 3

Пусть у нас k слоёв, тогда

2^(1/2)\*2^(1/4)\*…\*2^(1/2^(k-1)) \* n^(1/2^(k-1)) = 3

2^(1 – ½^(k-1)) \* n^(1/2^(k-1)) = 3

(n/2)^(1/2^(k-1)) = 3/2

(n/2)^(1/2^(k) \* 2) = 3/2

(n/2)^(1/2^(k)) = sqrt(3/2)

таких слоёв будет (log2(log\_{sqrt(3/2)}(n/2)))

Теперь осталось понять, какая суммарная затрата на возврат функций.

На 1-м слое кол-во функций составляет 1, а возврат каждой из них – sqrt(n).

Суммарный возврат слоя равен 1\*sqrt(n)\*c, где c = 1+sqrt(2) = const (будет использоваться в следующих подсчётах)

На 2-м слое кол-во функций составляет sqrt(2n) , а возврат каждой из них – sqrt(sqrt(2n)).

Суммарный возврат слоя равен sqrt(2n)\*sqrt(sqrt(2n))\*c = sqrt(2n)\*sqrt(2n)/sqrt(sqrt(2n))\*c = 2n / sqrt(sqrt(2n))\*c.

На 3-м слое кол-во функций составляет sqrt(2n) \* sqrt(2sqrt(2n)), а возврат каждой из них – sqrt(sqrt(2sqrt(2n))).

Суммарный возврат слоя равен sqrt(2n)\*sqrt(2sqrt(2n))\*sqrt(sqrt(2sqrt(2n)))\*c = sqrt(2n)\*sqrt(2sqrt(2n))\*sqrt(2sqrt(2n))/sqrt(sqrt(2sqrt(2n)))\*c = sqrt(2n)\*2sqrt(2n)/sqrt(sqrt(2sqrt(2n))) = 4n/sqrt(sqrt(2sqrt(2n)))\*c.

…

На k-м слое кол-во функций составляет sqrt(2n) \* sqrt(2sqrt(2n)) \* … \* sqrt(2sqrt(2sqrt(…2sqrt(2n))), а возврат каждой из них – sqrt(sqrt(2sqrt(2sqrt(…2sqrt(2n)))).

Также скроем все корни, как и раньше и получим ответ суммарного возврата слоя, как 2^(k-1)\*n/sqrt(sqrt(2sqrt(2sqrt(…2sqrt(2n))))\*c.

Рассмотрим знаменатель:

2^(1/4)\*2^(1/8)\*2^(1/16)\*…\*2^(1/2^k) \* n^(1/2^k)\*c = 2^(1/2-1/2^k) \* n^(1/2^k)

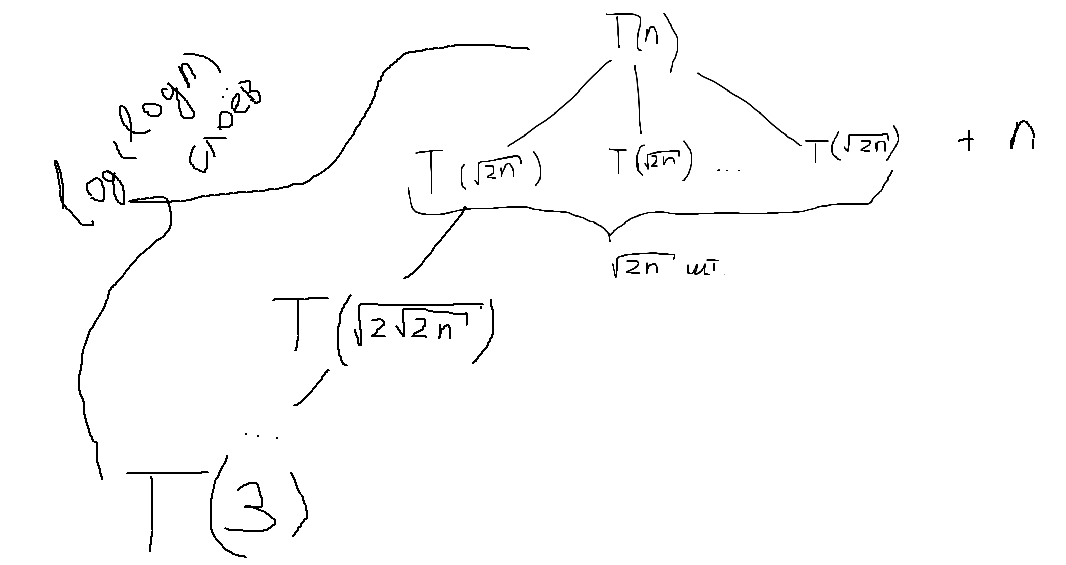
Значит общей дроби соответствует такое выражение: 2^(k-3/2+1/2^k) \* n^(1–1/2^k)

Заметим, что на k-ом слое суммарный возврат как минимум в 2 раза дольше, чем на

k-1, значит общая асимптотика совпадает с асимптотикой последнего слоя.

Найдём общую асимптотику (подставим k = log2(log3(n))):

1. 



Всё, как и в прошлом примере, только теперь у нас другое время возврата на слоях.

Заметим, что на каждом нижнем слое в 2 раза большая сумма аргументов, чем на предыдущем, так же время возврата равно сумме аргументов при вызовах.

То есть асимптотика всей рекурсии совпадает с асимптотикой суммы аргументов.

Последний слой содержит 2^(log2(log\_{sqrt(3/2)}(n/2))) \* n = log\_{sqrt(3/2)}(n/2)) \* n, что чуть больше половины всего времени.

Ответ:

1. Анализ последовательности

A[0] = 2

A[1] = 2

A[n] = 2\*A[n-1] + 2\*A[n-2]

Получим характеристическое уравнение, путём подставления r^k вместо A[k] в уравнении рекурсивного перехода:

r^n = 2\*r^(n-1) + 2\*r^(n-2)

r^(n-2) \* (r^2 – 2r - 2) = 0

D = 4 + 8 = 12

Нас интересуют ненулевые корни этого уравнения, то есть

Значит A[n] = a\*(r1)^n + b\*(r2)^n

Найдём a и b:

A[0] = a + b = 2

A[1] = a \* r1 + b \* r2 = 2

Значит a = 1, b = 1.

То есть

Проверка:

A[0] = 1 + 1 = 2 – верно.

A[1] = 1 + sqrt(3) + 1 – sqrt(3) = 2 – верно.

A[n] = 2\*A[n-1] + 2\*A[n-2] = 2 \* (1 + sqrt(3))^(n-1) + 2 \* (1-sqrt(3))^(n-1) + 2 \* (1 + sqrt(3))^(n-2) + 2 \* (1-sqrt(3))^(n-2) = (1 + sqrt(3))^(n-2) \* (2+2sqrt(3) + 2) + (1 - sqrt(3))^(n-2) \* (2 – 2sqrt(3) + 2) = (1+sqrt(3))^n + (1-sqrt(3))^n. Верно!

Поиск sqrt(3) можно осуществлять через эту последовательность, т. к.

Значит можно вычислять sqrt(3), как предел отношения двух членов этой последовательности минус единица.

Чем дальше мы найдём члены последовательности, тем ближе к реальному значению сможем вычислить корень из трёх.

Чтобы найти корень из k, нам нужно, чтобы дискриминант был равен 4k.

Пусть последовательность такова: A[1] = A[2] = 1; A[n] = x\*A[n-1] + y\*A[n-2].

То есть квадратное уравнение в характеристическом уравнении будет: r^(n-2) \* (r^2 – r\*x – y) = 0

D = x^2 + 4y = 4k => k = y + (x/2)^2

Можно выбрать пару x = 2, y = k-1.

A[n] = 2\*A[n-1] + (k-1)\*A[n-2].

То есть

Поиск sqrt(k) можно осуществлять через эту последовательность, т. к.

Значит можно вычислять sqrt(k), как предел отношения двух членов этой последовательности минус единица.

Это работает для всех натуральных k.