ТЕСТ 2. 18.05.2015 ◇

ВАРИАНТ 1

1	2	3	4	5	6	7	8	9	10	11	12	13	14	$\sum$

Фамилия, имя Группа

Задача 1. Да Нет Верно ли, что задача выполнимости КНФ остаётся NP-полной, если потребовать дополнительно, что в каждом дизъюнкте есть литерал, такой что отвечающая ему переменная равна 1? *Пример:* $(\neg x_1 \vee \neg x_2 \vee \neg x_3 \vee x_4) \wedge (\neg x_1 \vee x_2) \wedge (\neg x_2 \vee \neg x_3) \wedge (x_3 \vee x_4)$. Набор $(0, 1, 0, 1)$ выполняет КНФ, и при этом в каждом дизъюнкте есть переменная, которая равна 1.

$\diamond$

Задача 2. Да Нет Докажите или опровергните NP-полноту проверки выполнимости КНФ, в которых в каждый дизъюнкт входит не более одной переменной без отрицания.
Пример: $(\neg x_1 \vee x_2 \vee \neg x_3 \vee \neg x_2) \wedge (\neg x_1 \vee \neg x_3)$.

◇

Задача 3. Да Нет Верно ли, что язык, состоящий из описания графов, в которых каждый цикл, состоящий из $|V|$ рёбер, проходит через некоторую вершину минимум два раза, является полиномиально полным в классе $co - \mathcal{NP}$?

$\diamond$

Задача 4. Да Нет Верно ли, что язык $L = \{\langle G, s, t\rangle \mid G$ —- описание ориентированного графа; $s, t \in V(G)$; в G есть ровно два реберно непересекающихся пути из s в $t\}$ является полиномиально полным в классе NP?

◇

Задача 5. Да Нет Верно ли, что связный граф, содержащий не менее трех вершин, является двусвязным, если в нем нет точек раздела?

$\diamond$

Задача 6. Да Нет В ориентированном связном графе G запустили поиск в глубину и вычислили значения параметров d и f для каждой вершины графа. Верно ли, что компонента сильной связности B достижима из компоненты сильной связности A тогда и только тогда, когда $\forall a \in A \ \exists b \in B : d[a] > d[b], \ f[a] > f[b]$. В случае отрицательного ответа приведите контрпример.

$\diamond$

Задача 7. Вычислите $19^{12^{117973}} \mod 11$.

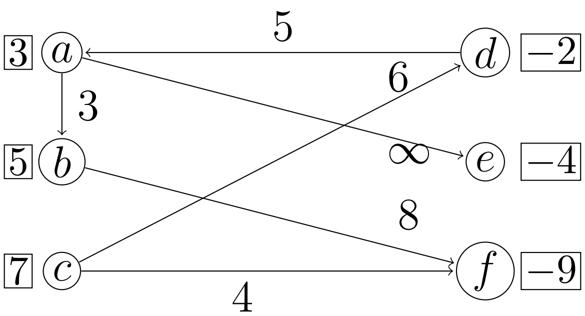
Задача 8. Решите уравнение $x^8 \equiv 100 \pmod{31}$. Известно, что на отрезке $[2, 5]$ есть первообразный корень $\pmod{31}$.

◇

Задача 9. Да Нет Массив из 10000 различных чисел сортируется посредством вероятностного алгоритма быстрой сортировки QUICKSORT. Верно ли, что существует такой прогон алгоритма, при котором числа, занимающие по величине 2015-е и 2017-е место, не будут сравниваться между собой.

$\diamond$

Задача 10. Пусть $G = (V, A)$ — сеть с заданными пропускными способностями ребер. Кроме того, каждой вершине $i \in V$ присвоен вес $b(i)$. Вес вершины можно рассматривать как запас, если $b(i) > 0$ или как спрос, если $b(i) < 0$, при этом сумма всех весов вершин сети равно нулю. Поток называется допустимым, если разность между притоком и истоком из каждой вершины равна весу этой вершины. В отличие от обычного закона сохранения потока, теперь в сумму включается не только входящий и исходящий поток, но и вес вершины $b(i)$. Предложите полиномиальный алгоритм, который по входу задачи определяет, существует ли в сети допустимый поток. Продемонстрируйте работу вашего алгоритма на примере следующего графа (в прямоугольниках записаны веса вершин):



$\diamond$

Задача 11. На схеме приведена блок схема вычисления функции БЕДНЫЙ $(\cdot)$, которая использует вызов функции СТЯГИВАНИЕ $(\cdot)$[1]. Известно, что трудоемкость вычисления функции СТЯГИВАНИЕ(n) равна $\Theta(n^2)$.

БЕДНЫЙ(n)
if $n = 2$ then БЕДНЫЙ$(2) = 1$
else
СТЯГИВАНИЕ (n)
$X_1 \leftarrow$ БЕДНЫЙ$(\lceil n/\sqrt{2} \rceil + 1)$;
СТЯГИВАНИЕ (n)
$X_2 \leftarrow$ БЕДНЫЙ$(\lceil n/\sqrt{2} \rceil + 1)$;
return $\min\{X_1, X_2\}$

Запишите рекуррентную оценку сложности $T(n)$ вычисления БЕДНЫЙ(n) (0.5 балла) и найдите Θ-асимптотику $T(n)$ (3 балла).

Если рекуррентность для $T(n)$ неверная И/ИЛИ если не анализируются эффекты, связанные с использованием иррациональности, функции $\lceil \cdot \rceil$ и сдвигом аргумента, то этот пункт оценивается из 1.5 баллов.

[1]Использование странных идентификаторов станет понятно, если прочитать комментарий после следующей задачи, но это совершенно не обязательно.

$\diamond$

Задача 12. Приведите как можно более точную оценку снизу рекурсии: $p_0 = 1; p_{k+1} = p_k - \frac{3}{8}(p_k)^2$ вида $p_k = \Omega(f(k))$.

Подсказка. Предположим, что $p_{k+1} - p_k \approx \frac{dp}{dk}$, и оценим порядок роста функции $p(k)$, а потом обоснуем нашу гипотезу.

◇

Комментарий не для контрольной, в котором говорится, откуда взялась рекурсия в № 12, и почему в № 11 используются странные идентификаторы. В задаче №54 задания рассмотрен вероятностный алгоритм поиска минимального разреза, предложенный Д. Каргером (“karg” (нем.) — бедный) двадцать лет назад. Алгоритм последовательно выбирает случайные ребра и стягивает их концы до тех пор, пока в графе не останутся две вершины, соединенных (кратными) ребрами. Если при выборе случайных ребер мы ни разу не выбирали ребра разреза, то мы получаем ответ. В задаче №54 показывается, что вероятность на i-м шаге выбрать ребро, входящее в разрез, равна $p_i = \frac{2}{n-i}$, отсюда вероятность того, что за i шагов не будет выбрано ни одно ребро, входящее в минимальный разрез (мы будем говорить, что выбранные *ребра не задевает разрез*) равна $P_i = (1-p_1)(1-p_2)\dots(1-p_i)$. Мы хотим ускорить алгоритм. Заметим, что чем больше ребер мы стягиваем, тем больше вероятность, что следующее выбранное ребро заденет минимальный разрез. Поэтому новая идея, которую мы хотим исследовать, заключается в том, чтобы стягивать ребра до какого-то порога, пока вероятность попадания в разрез еще достаточно мала, а дальше использовать рекурсию. Из формулы для P_i видно, что если стянуть $n/2$ случайных ребер, то они с вероятностью $\geq \frac{1}{4}$ не заденут минимальный разрез.

Обозначим $G \leftarrow G/e$ операцию стягивания ребра e в графе G (петли удаляются)

Блок-схема нового алгоритма приведена ниже. Процедура использует подпрограмму СТЯГИВАНИЕ (G, k), которая стягивает ребра до тех пор пока число вершин не уменьшится ниже порога k (напомним, что при стягивании произвольного ребра число вершин уменьшается на единицу).

n обозначает число вершин в (текущем) графе G.

```
СТЯГИВАНИЕ (G, k)
for i := n downto k
В G выбираем случайное ребро
e
(с равномерным распределени-
ем
на ребрах).
G ← G/e
endfor
return G.
```

```
МИН-РАЗРЕЗ (G)
if в G больше восьми вершин then
Повторим 4 раза процедуру
X_1 ← МИН-РАЗРЕЗ [СТЯГИВАНИЕ (G, n/2)];
X_2 ← МИН-РАЗРЕЗ [СТЯГИВАНИЕ (G, n/2)];
X_3 ← МИН-РАЗРЕЗ [СТЯГИВАНИЕ (G, n/2)];
X_4 ← МИН-РАЗРЕЗ [СТЯГИВАНИЕ (G, n/2)];
return min{X_1, X_2, X_3, X_4}
else находим минимальный разрез вручную.
```

Обратите внимание, что если в рекурсии использовать двухкратное повторение, то программа будет очень похожа на процедуру БЕДНЫЙ из № 11.

Оценим теперь с какой вероятностью $\mathbb{P}(n)$ алгоритм МИН-РАЗРЕЗ(G) выдает минимальный разрез для графа G с n вершинами. Вероятность успеха равна вероятности того, что хотя бы один рекурсивный вызов дал корректный ответ. Таким образом, получаем: $\mathbb{P}(n) = 1 - \left(1 - \frac{1}{4}\mathbb{P}(\frac{n}{2})\right)^4$, откуда следует, что $\mathbb{P}(n) \geq \mathbb{P}(\frac{n}{2}) - \frac{3}{8}\mathbb{P}(\frac{n}{2})^2$. Считая, что n является степенью двойки, обозначим $p_k = \mathbb{P}(2^k)$. По определению, p_k равно вероятности успеха, если потребовалось k рекурсивных вызовов процедуры МИН-РАЗРЕЗ. В частности, $p_0 = 1$.

Получаем рекурсию $p_{k+1} = 1 - (1 - \frac{1}{4}p_k)^4$. Откуда, если раскрыть скобки, следует: $p_{k+1} \geq p_k - \frac{3}{8}(\mathcal{P}_k)^2$. Таким образом оценка рекурсии в предыдущей задаче — это оценка трудоемкости модифицированно вероятностного алгоритма поиска минимального разреза, откуда можно получить оценку на число итераций алгоритма для получения заданной точности ε

$\diamond$

Задача 13. Как мы знаем из курса ТРЯП, число правильных скобочных выражение длины $2k$ равно числу Каталана c_k. А из курса дискретного анализа мы знаем, что производящая функция чисел Каталана равна $f(t) = \frac{1-\sqrt{1-4t}}{2t}$. Найдите явное выражение для производящей функции $\varphi_1(t) = t + c_4t^4 + c_7t^7 \dots$ скобочных выражение длины $6k+2$, $k = 0, 1 \dots$ через $f(t)$.

Например, соответствующее представление для скобочных выражений длины $4k$, $k = 0, 1, \dots$ равно $\frac{f(t)+f(-t)}{2}$, а для скобочных выражений длины $4k+2$, $k = 0, 1, \dots$ — $\frac{f(t)-f(-t)}{2}$.

Подсказка. ± 1 — это корни из единицы.

$\diamond$

Задача 14. Пусть $G = (V, E)$ — неориентированный граф с весами на ребрах. Назовем min-весом пути минимум из весов ребер этого пути. Приведите как можно более быстрый алгоритм, алгоритм, определяющий min-вес между всеми парами вершин графа при условии, что веса могут быть и отрицательными.