Филиппова Анастасия

Домашнее задание 3

1. Вариант 1: Сведем язык 3-ВЫПОЛНИМОСТЬ (буду обозначать A) к языку РОВНО-3-ВЫПОЛНИМ (буду обозначать B). Пусть $x \in A$, где x — выполнимая КНФ, что в каждом дизъюнкте не более трех литералов (то есть существует набор, на котором она принимает значение 1). Придумаем такую функцию полиномиальную по длине входа, что $x \in A \iff f(x) \in B$. x заменим на КНФ, в которой нет повторяющихся переменных, то есть $(a \vee a \vee b) = (a \vee b)$. Пусть $a_1 \vee ... \vee a_k \vee b_1 \vee ... \vee b_m \vee c_1 \vee ... \vee c_n = x -$ замененная кнф (то есть нив одном дизъюнкте нет повторяющихся литералов), a_i — дизъюнкты с одним литералом, b_i — с двумя,c_i — с тремя. Модифицируем дизъюнкты a_i и b_i следующим образом (дополним их так, чтобы каждый дизъюнкт содержал 3 литерала и сохранялась бы выполнимость):
   Введем дополнительные переменные u, v, u_1, u_2, v_1, v_2:
   $A_i = (a_i \vee u \vee v) \wedge (\bar{u} \vee u_1 \vee u_2) \wedge (\bar{u} \vee \bar{u}_1 \vee u_2) \wedge (\bar{u} \vee \bar{u}_1 \vee \bar{u}_2) \wedge (\bar{u} \vee u_1 \vee \bar{u}_2) \wedge (\bar{v} \vee v_1 \vee v_2) \wedge (\bar{v} \vee \bar{v}_1 \vee v_2) \wedge (\bar{v} \vee \bar{v}_1 \vee \bar{v}_2) \wedge (\bar{v} \vee v_1 \vee \bar{v}_2)$.
   $(\bar{u} \vee u_1 \vee u_2) \wedge (\bar{u} \vee \bar{u}_1 \vee u_2) \wedge (\bar{u} \vee \bar{u}_1 \vee \bar{u}_2) \wedge (\bar{u} \vee u_1 \vee \bar{u}_2) \wedge (\bar{v} \vee v_1 \vee v_2) \wedge (\bar{v} \vee \bar{v}_1 \vee v_2) \wedge (\bar{v} \vee \bar{v}_1 \vee \bar{v}_2) \wedge (\bar{v} \vee v_1 \vee \bar{v}_2)$ выполнима тогда и только тогда, когда $u = v = 0$, поэтому A_i выполнима тогда и только тогда, когда выполнима КНФ a_i. Для b_i проведем похожие модификации: $B_i = (b_i \vee z) \wedge (\bar{z} \vee m \vee n) \wedge (\bar{z} \vee \bar{m} \vee n) \wedge (\bar{z} \vee \bar{m} \vee \bar{n}) \wedge (\bar{z} \vee m \vee \bar{n})$ — выполнима тогда и только тогда,когда $z = 1$ и выполнима b_i. Модифицированная КНФ $A_1 \wedge ... \wedge A_k \vee B_1 \vee ... \vee B_m \vee c_1 \vee ... \vee c_n$ выполнима тогда и только тогда, когда выполнима исходная КНФ и в каждом дизъюнкте содержит ровно 3 литерала. Очевидно, что так как мы добавляем конечное число дизъюнктов, алгоритм работает за полиномиальное время от длины входа.
   Вариант 2: Можно сделать в проще и просто дополнить каждый дизъюнкт, не меняя его выполнимость, то есть :$(x_1 \vee x_2) \to (x_1 \vee x_2 \vee x_1)$

2. Построить сводимость SAT к языку ПРОТЫКАЮЩЕЕ МНОЖЕСТВО.

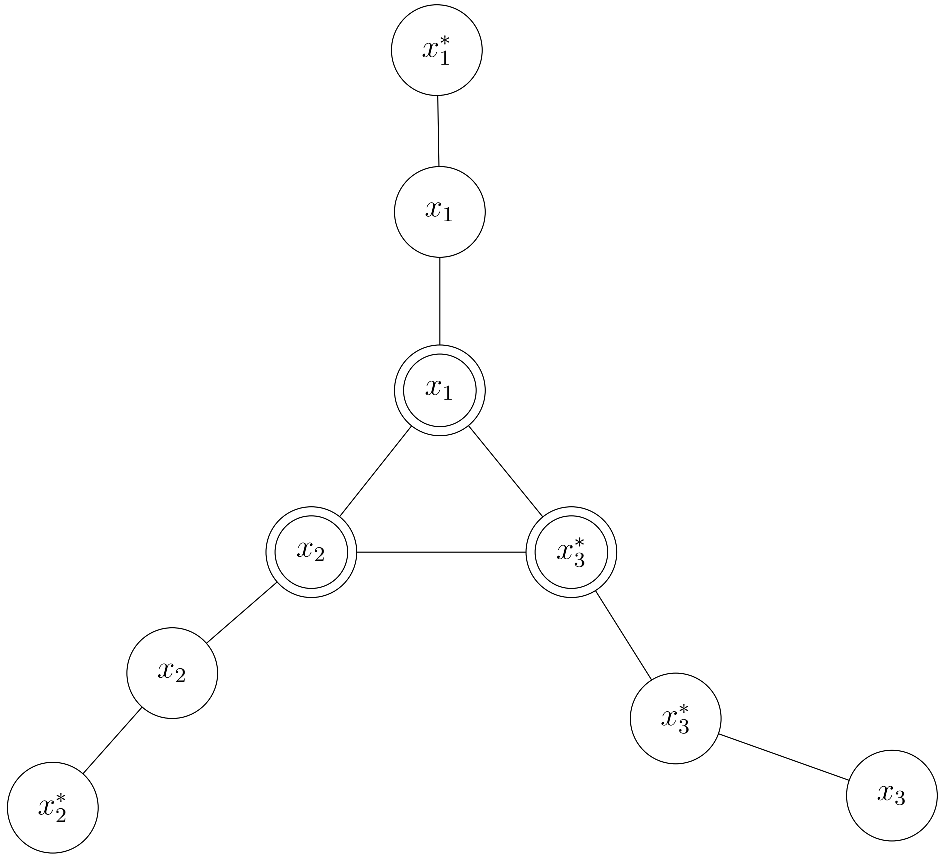
   (a) $\psi(x_1, x_2, x_3) = (x_1 \vee x_2 \vee \bar{x}_3)$. Протыкающее множество для семейства A_ψ подмножеств множества $\{x_1, x_1, x_3, \bar{x}_1, \bar{x}_2, \bar{x}_3\}$, построенного по алгоритму из методички есть множество $B = \{x_1, x_2, \bar{x}_3\}$. Доказательство очевидно из построения семества подмножеств.

   (b) $\chi = (x_1 \vee x_2) \wedge (x_1 \vee \bar{x}_2) \wedge \bar{x}_1$. Докажем, что у семейства подмножеств A_χ множества $\{x_1, x_2, \bar{x}_1, \bar{x}_2\}$ все протыкающие множества имеют мощность больше 2. Семейство содержит множество $\{\bar{x}_1\}$, следовательно, $\bar{x}_1 \in$ протыкающее множество. В семействе 3 множества, не содержащие $\bar{x}_1 : \{x_1, x_2\}, \{x_1, \bar{x}_2\}, \{x_2, \bar{x}_2\}$. Эти множества пересекаются по двум элементам, следовательно, протыкающее множество не может иметь мощность меньше чем 2.

   Сводимость: покажем,что КНФ выполнима тогда и только тогда, когда у построенного по алгоритму семейства существует протыкающее множество мощности n. Пусть есть КНФ $\alpha = \alpha(x_1, x_2, ..., x_n)$, построим A_α по алгоритму. В следствии построения имеем, что КНФ выполнима тогда и только тогда, когда найдется набор переменных такой, что в каждом множестве семейства будет хотя бы одна 1. Пусть КНФ выполнима, следовательно, в любом множестве из построенного семейства есть хотя бы одна 1. Построим протыкающее множество следующим образом: поместим в него элементы, обращающиеся в единицу на наборе, на котором КНФ выполнима. Дополним этот набор так, чтобы полученное множество было протыкающим. Пример: рассмотрим КНФ ψ. Она выполнима на наборе $1, 0, 1. A_{1,2,3} = \{x_{1,2,3}, \bar{x}_{1,2,3}\}, A_4 = \{x_1, x_2, \bar{x}_3\}$. Добавляем x_1 (единицу из A_4), далее дополняем до протыкающего, добавляя $x_2, \bar{x}_3$. Мощность построенного множества n. Действительно, из A_i берем от элемент, который обращается в 1 (если $\bar{x}_1 = 1$, помещаем в протыкающее множество $\bar{x}_1$). Очевидно, что на этом шаге будет не более n элементов в построенном множестве. Далее дополняем так, чтобы были пересечания со всеми n множествами, содержащими переменную и ее отрицание. В обратную сторону: пусть есть протыкающее множество мощности n, по обратному алгоритму строим КНФ. Докажем, что она будет выполнима. Аналогично доказательству, написанному выше: помещаем единицы в элементы из протыкающего множества. Очевидно, что КНФ будет выполнима. Доказано.

3. Построить сводимость языка $3-SAT$ к языку ВЕРШИННОЕ ПОКРЫТИЕ.

   (a) Вершинное покрытие для графа, изображенного ниже ($x_2^* = \bar{x}_2$, вершина в двойном кружке — дизъюнктная вершина): $x_2, x_{2D}, x_1, x_{3D}^*, x_3^*$, где x_{2D} — дизъюнктная вершина.



   (b) Смотреть доказательство ниже.

   Покажем, что КНФ выполнима тогда и только тогда, когда в графе, построенного по алгоритму есть вершинное покрытие мощности $n + 2m$.
   Пусть КНФ не выполнима, значит, существует такой набор, что в каком-то дизъюнкте после подстановке не будет ни одной 1. Это значит, что, выражаясь на языке графов, в построенном для данной КНФ будет треугольник (дизъюнктный) такой, что ни одно ребро, выходящее из этого дизъюнктного треугольника не будет покрыто литеральной вершиной (потому что смежные литеральные вершины — нули, в покрытие добавляем 1). В обратную сторону аналогично: пусть в графе нет вершинного покрытия размера $n + 2m$, это значит, что выбрав по одной вершине для покрытия ребер вида $\{x_i, \bar{x}_i\}$ и по 2 ребра для дизъюнктных треугольников, мы не придем к успеху, то есть не покроем все ребра, следовательно, будет дизъюнктный треугольник, что ребра, выходящие из его вершин при любом выборе литеральных вершин, входящих в вершинное покрытие графа, не будут покрыты литеральными вершинами.

   В общем, выбираем n литеральных вершин (x_i, если $x_i = 1$, в противном случае $\bar{x}_i$). Далее выбираем $2m$ дизъюнктных вершин, чтобы покрыть оставшиеся ребра. Делаем так, чтобы в каждом конъюнкте была хотя бы одна 1, то есть для любого дизъюнктного треугольника хотя бы одно ребро из нео выходящее покрыто литеральной вершиной. Для КНФ χ мы этого сделать не можем, как бы мы не дополняли ее до РОВНО-3-КНФ, потому что, очевидно, что нужно хотя бы 8 вершин для покрытия, но ровно 8 быть не может, так как КНФ не выполнима, а выше показано, что если КНФ не выполнима, то нельзя построить такое покрытие.

4. Построить сводимость языка РОВНО-3-ВЫПОЛНИМОСТЬ к языку CLIQUE.

(a) Очевидно, граф представляет собой 3 не смежные с друг другом вершины. Очевидно, что клика размера $m = 1$ найдется.

(b) Надо найти клику размера 3. По описанному алгоритму понятно, что клика может быть не больше чем размера 3 (потому что вершины из одинаковых дизъюнктов не соединены ребрами). То есть, грубо говоря, надо выделить треугольник. Но для такой КНФ он никогда не получится, как бы мы ее не дополняли до ровно 3−КНФ.

Пусть КНФ выполнима. Построим граф по алгоритму. Существует такой набор, что каждом дизъюнкте есть хотя бы одна 1. Те вершины, которые обращаются в 1 соединены!(ведь мы не соединяем только вершину и ее отрицание), следовательно, те вершины, которые будут обращаться в 1 будут образовывать клику размера m, где m — число дизъюнктов. В общем, доказательство аналогично доказательству, приведенному на семинаре, только не для РОВНО-3-ВЫПОЛНИМОСТИ,а для ВЫПОЛНИМОСТИ.

5. Построим сводимость языка 3−ВЫПОЛНИМОСТЬ к языку MAX-2-ВЫПОЛНИМОСТЬ.

(a) $\psi = x_1 \vee x_2 \vee \bar{x}_3$, вычислим 2-КНФ по указанному алгоритму: $\widetilde{\psi} = x_1 \wedge x_2 \wedge \bar{x}_3 \wedge x_4 \wedge (\bar{x}_1 \vee \bar{x}_2) \wedge (\bar{x}_1 \vee x_3) \wedge (\bar{x}_2 \vee x_3) \wedge (x_1 \vee \bar{x}_4) \wedge (x_2 \vee \bar{x}_4) \wedge (\bar{x}_3 \vee \bar{x}_4)$ (добавляю 3 дизъюнкции вида $(\bar{a}_i \vee \bar{b}_i)$, сами переменные, дополнительную переменную и 4 дизъюнкта (дизъюнкцию каждой переменной с новой). Итог: 10 новых дизъюнктов. Заменим в построенной КНФ $\bar{x}_3$ на у для удобства и рассмотрим общий случай. КНФ принимает вид:$\widetilde{\widetilde{\psi}} = x_1 \wedge x_2 \wedge y \wedge x_4 \wedge (\bar{x}_1 \vee \bar{x}_2) \wedge (\bar{x}_1 \vee \bar{y}) \wedge (\bar{x}_2 \vee \bar{y}) \wedge (x_1 \vee \bar{x}_4) \wedge (x_2 \vee \bar{x}_4) \wedge (y \vee \bar{x}_4)$.
1) Случай: одна переменная принимает значение 1 (в силу абсолютной симметричности с точки зрения переменных, неважно, какая именно). Пусть x_1. Не трудно посчитать, что 7 дизъюнктов и не больше принимают значение 1 ($x_4 = 1$).
2) Случай: 2 переменные принимаю значение 1 (не умаляя общности, скажем, что это x_1, x_2). $x_1, x_2, \bar{x}_1 \vee \bar{y}, \bar{x}_2 \vee y$ обращаются в 1. Подберем x_4 так, чтобы максимизировать это количество. Очевидно, что при $x_4 = 0$ и при $x_4 = 1$ количество истинных дизъюнктов будет одинаково и равно 7.
3) Случай: все переменные 1. Истинные дизъюнкты: $x_1, x_2, y, x_1 \vee \bar{y}, x_2 \vee \bar{y}, x_3 \vee \bar{y}$, подобрав $x_4 = 1$, получаем максимальное количество 7. Следовательно, 7 — пороговое значение.

(b) Набор $1, 1, 0, 1$, выполняется 7 дизъюнктов.

Пусть КНФ не выполнима, значит в исходной КНФ все переменные принимают значение 0. Тогда, очевидно, что единичные дизъюнкты обращаются в 0 (x_1, x_2, y), истинной значение принимают дизъюнкты вида:$(\bar{a}_i \vee \bar{b}_i)$ — 3 штуки. И далее, если $x_4 = 1$ из оставшихся истинен только 1. Если $x_4 = 0$, то истины 3. Получаем 6. В обратную сторону аналогично: если из 10 дизъюнктов выполнимо меньше 7, то КНФ не выполнима.
Следовательно, произвольная 3−КНФ будет выполнима тогда и только тогда, когда в построенной $max - 2-$КНФ не меньше чем $7k$ дизъюнктов будет выполнимо.
Алгоритм сведения полиномиальный, так как КНФ полиномиально сводится к 3−КНФ (первая задача). Далее каждый дизъюнкт заменяем на конъюнкцию 10 дизъюнктов. Очевидно, что в 3−КНФ дизъюнктов полиномиальное число от длины входа (предполагая под длиной входа исходную КНФ). Следовательно, алгоритм полиномиален.

6. На семинаре построили сводимость языка $3 - SAT$ к языку $COLOR$. Так как КНФ выполнима тогда и только тогда, когда граф можно правильно раскрасить в 3 цвета (граф, построенный по алгоритму с семинара), следовательно, докажем, что если $3 - SAT \in P$, то за полиномиальное время можно не только определить является формула выполнимой или нет, но и найти набор, на котором она выполняется. Будем поочереди проверять все переменные следующим образом: пусть $\psi(x_1, x_2, ..., x_n)$ — КНФ, в каждом дизъюнкте которой не более трех литералов. Подставляем x_1, равное 1. При этом из КНФ удаляем все дизъюнкты, содержащие x_1 и удаляем из всех дизъюнтов $\bar{x}_1$ (очевидно). Далее полученную КНФ $\psi(x_2, x_3, ..., x_n)$ проверяем на выполнимость за полиномиальное время (по условию задачи). Если формула выполнима, делаем такие же действия с переменной x_2, если нет — подставляем $x_1 = 0$ и также проверяем

КНФ $\psi(x_2, x_3, ..., x_n)$ на выполнимость за полином. Время работы алгоритма в худшем случае: пусть алгоритм распознования 3_SAT работает за $O(poly(n))$, тогда алгоритм нахождения такого набора, на котором формула выполнима, работает за $2 \cdot n \cdot O(poly(n)) = O(poly(n))$. Следовательно, за полиномиальное время можем найти набор, на котором формула выолнима, то есть за полиномиальное время можем найти рассkраску графа.

7.