Домашняя работа 3 — Кузнецов

1 Задание 3

1. C21. Изоморфен ли граф G_2 подграфу графа G_1 — NP полна. Задача о независимом множестве (NP-полнота которой была доказана на лекции) сводится к этой задаче. Действительно, возьмём в качестве графа G_2 граф на k вершинах без рёбер. Тогда задача об изоморфном подграфе эквивалентна вопросу о том, существует ли в графе G_1 независимое множетво размера k.

2. C22. Задача о выполнимости КНФ сводится к этой задаче. Действительно, допустим, у нас есть КНФ. Для каждой переменной x_i заведём переменную t_i. Все вхождения x_i без отрицания заменим на $\neg t_i$, и также добавим в конъюнкцию дизъюнкты $t_i \vee x_i$, $\neg t_i \vee \neg x_i$ для всех i. Покажем, что эта формула выполнима тогда и только тогда, когда выполнима исходная формула. Действительно, если исходная формула выполнима при некотором наборе $x_1, \dots, x_n$, то положив $t_i = \neg x_i$, получаем набор, при котором выполима новая формула. И наоборот, если новая формула выполнима, то из

$$(t_i \vee x_i) \wedge (\neg t_i \vee \neg x_i)$$

   следует, что $t_i = \neg x_i$, и ясно, что исходная формула выполнима, т.к. она получается из части новой формулы заменой всех t_i на $\neg x_i$.

3. C26. a) $NP \subseteq EXP$.
   Пусть L — некоторый язык из класса NP. Существует полиномиальный алгоритм P и многочлен p, такой, что $x \in L$ тогда и только тогда, когда существует двоичное слово w длины не более $p(n)$, где n — длина x, такое, что $P(x, w) = 1$. Тогда для определения принадлежности слова x языку L мы можем просто перебрать все слова w длины не более $p(n)$, которых не более $2^{p(n)+1}$, за экспоненциальное время.

Поэтому L лежит в EXP — он распознаётся за экспоненциальное время.

б) NP замкнут относительно сведений по Карпу.
Если L — язык из NP, и L_1 сводится по Карпу к L, то L_1 лежит в NP. Докажем это. Из принадлежности L классу NP ясно, что существует полиномиальный алгоритм P и многочлен p, такой, что $x \in L$ тогда и только тогда, когда существует двоичное слово w длины не более $p(n)$, где n — длина x, такое, что $P(x, w) = 1$. Из сводимости следует, что существует полиномиально вычислимая функция f, такая, что $x \in L_1$ тогда и только тогда, когда $f(x) \in L$. Тогда построим алгоритм $H, H(x, w) = P(f(x), w)$. Он и будет проверяющим алгоритмом для L_1. Подсказкой для x будет то же слово w, что в L служило подсказкой для $f(x)$.

4. C27. Пример EXP-полной задачи.
Построим EXP-полный язык H. Этот язык будет состоять из троек $(M, x, 1^t)$, где M — машина Тьюринга (её описание), таких, что M выдаёт ответ Yes на входе x за не более чем 2^t шагов. Этот язык лежит в EXP, поскольку проверить принадлежность тройки языку можно за экспоненциальное время. Покажем, что любой другой язык из EXP сводится к этому языку. Допустим, у нас есть язык L из класса EXP, для которого существует машина Тьюринга A, которая распознаёт его за время, меньшее $2^{p(n)}$, где p—многочлен. Сопоставим слову x тройку $f(x) = (A, x, 1^{p(n)})$ (n — длина слова x). Эта тройка вычислима за полиномиальное от длины x время. Ясно, что $f(x) \in H$ тогда и только тогда, когда $x \in L$. Мы свели язык L к H.

5. C6. а) Сдано в первом задании.

б) Плользуясь пунктом а), построим по исходной схеме C вероятностную схему C' размера $poly(s, n)$ с m' случайными битами для $p(n) = 2n$. Покажем, что найдётся детерминированная строка r, такая, что $f(x) = C'(x, r)$ для всех x. Действительно, для каждого фиксированного x вероятность того, что $f(x) \neq C'(x, r)$, не превосходит $2^{-p(n)}$. Тогда строк r, для которых $f(x) \neq C'(x, r)$, не более $2^{-p(n)}2^{m'}$. Значит, строк, для которых хотя бы для одного x выполнено $f(x) \neq C'(x, r)$, не более

$$2^n 2^{-p(n)} 2^{m'} = 2^{n-2n} 2^{m'} < 2^{m'},$$

то есть таких строк меньше, чем всего строк длины m', и найдётся строка длины m', для которой $f(x) = C'(x, r)$ при всех x. Заменив случайные биты этой строкой, получаем искомую детерминированную схему для f.

6. C17. а) Язык USAT coNP-трудный.
Язык выполнимых 3-SAT NP-полный, значит, язык невыполнимых 3-SAT coNP-полный. Сведём его к USAT. Пусть $F(x_1, \ldots, x_n)$ — некоторая $3-SAT$. Построим по ней формулу

$$A(x_1, \ldots, x_n) = (x_1 \wedge x_2 \wedge \ldots \wedge x_n) \vee F(x_1, \ldots, x_n) \vee F(\neg x_1, \ldots, \neg x_n).$$

Ясно, что F невыполнима тогда и только тогда, когда A имеет единственный выполняющий набор $x_1 = x_2 = \ldots = x_n = 1$, то есть лежит в USAT.

б) Если USAT лежит в UP, то NP=coNP.

Рассмотрим NP-полную задачу выполнимости 3-КНФ. Из того, что USAT лежит в UP, следует, что и невыполнимость 3-КНФ мы можем проверять с помощью полиномиальной подсказки (язык невыполнимых 3-КНФ сводится к USAT по предыдущей задаче, а для формулы из USAT по условию есть подсказка). В этом случае задача о невыполнимости 3-КНФ тоже лежит в класе NP. Это значит, что все языки из coNP, поскольку они сводятся к языку невыполнимых 3-КНФ, лежат в классе NP (класс NP замкнут относительно сведений по Карпу). Таким образом, класс coNP содержится в NP. А отсюда следует, что и наоборот, класс NP содержтся в классе coNP (берём любой язык L из класса NP, его дополнение лежит в coNP, значит, это дополнение лежит в NP, значит, сам язык L лежит в coNP).

В решении не использовалась единственность подсказки.