ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
МОСКОВСКИЙ ФИЗИКО-ТЕХНИЧЕСКИЙ ИНСТИТУТ
(НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ)

Алгоритмы и модели вычислений.
Домашняя работа 8.

Георгий Никишин
Группа Б05-909

Задача 1.
а) Докажите, что для любого $k \in \mathbb{N}$ выполнено $\Sigma_k \subset \pi_{k+1}$, $\pi_k \subset \Sigma_{k+1}$

Решение.
Рассмотрим определение обоих классов:
$\Sigma_k : x \in L \Leftrightarrow \exists R \exists y_1 \forall y_2 \exists y_3 \cdots : R(x,y_1,y_2,y_3 \cdots) = 1$
$\pi_{k+1} : x \in L \Leftrightarrow \exists R \forall y_1 \exists y_2 \forall y_3 \exists y_4 \cdots : R(x,y_1,y_2,y_3 \cdots) = 1$
Пусть $L \in \Sigma_k$, тогда $x \in L \Leftrightarrow \exists R \exists y_2 \forall y_3 \exists y_4 \cdots : R(x,y_2,y_3,y_4 \cdots) = 1$ пусть теперь R тот же предикат, но от k+2 переменных, причем 2-ая переменная фиктивная, то есть $\forall y_1 \Rightarrow R(x,y_1,y_2,y_3,y_4 \cdots) = 1 \Leftrightarrow R(x,y_2,y_3,y_4 \cdots) = 1$ тогда выполнено: $x \in L \Leftrightarrow \exists R \forall y_1 \exists y_2 \forall y_3 \exists y_4 \cdots : R(x,y_1,y_2,y_3 \cdots) = 1$, что и есть определение π_{k+1}
Аналогично рассмотрим $L \in \pi_k$, тогда: $x \in L \Leftrightarrow \exists R \forall y_2 \exists y_3 \forall y_4 \cdots : R(x,,y_2,y_3,y_4 \cdots) = 1$ добавим в R фиктивную переменную, тогда $x \in L \Leftrightarrow \exists R \exists y_1 \forall y_2 \exists y_3 \cdots : R(x,y_1,y_2,y_3 \cdots) = 1 \Rightarrow L \in \Sigma_{k+1}$
б) Покажите, что $\cup_{i \in \mathbb{N}} \Sigma_i = \cup_{i \in \mathbb{N}} \pi_i$.

Решение.
Рассмотрим произвольный язык $L \in \cup_{i \in \mathbb{N}} \Sigma_i \Rightarrow \exists i \in \mathbb{N} : L \in \Sigma_i$, тогда по предыдущему пункту $L \in \pi_{i+1} \Rightarrow L \in \cup_{i \in \mathbb{N}} \pi_i$, тогда $\cup_{i \in \mathbb{N}} \Sigma_i \subset \cup_{i \in \mathbb{N}} \pi_i$
Пусть теперь $L \in \cup_{i \in \mathbb{N}} \pi_i$, аналогично $\exists i \in \mathbb{N} : L \in \pi i \Rightarrow L \in \Sigma_{i+1} \Rightarrow L \in \cup_{i \in \mathbb{N}} \Sigma_i \Rightarrow \cup_{i \in \mathbb{N}} \pi_i \subset \cup_{i \in \mathbb{N}} \Sigma_i$, тогда $\cup_{i \in \mathbb{N}} \Sigma_i = \cup_{i \in \mathbb{N}} \pi_i$.

Задача 3.
Докажите, что PSPACE $\subset$ EXPTIME.

Решение.
Рассмотрим, какое время может занимать работа МТ для распознавания языка из PSPACE. Тк у нас есть полиномиальное число ячеек памяти $p(n)$, то на ленте может быть $2^{p(n)}$ различных строк из 0 и 1. Пусть у нашей МТ k возможных положений головки, тогда всего комбинаций из состояний и положения головки: $T(n) = p(n) \cdot k \cdot 2^{p(n)}$. Тогда если при работе данной МТ будет хотя бы $T(n) + 1$ иттерация, то по принципу Дирихле мы должны посетить одну из данных конфигураций хотя бы дважды, тк МТ детерменирована то при этом она, очевидно, зациклится. Таким образом, получаем противоречие. Отсюда следует вывод, что любой язык из PSPACE распознается за $T(n) = p(n) \cdot k \cdot 2^{p(n)} \leq 2^{q(n)}$, где $q(n)$ - некоторый полином. Таким образом, PSPACE $\subset$ EXPTIME.

Задача 4б.
Показать, что UCYCLE в L.

Решение.
Воспользуемся алгоритмом поиска в ширину для решения этой задачи. Тк наш граф неориентированный, то одно ребро не должно встречаться дважды . Поэтому необходимо дополнительно проверять, что текущее рассматриваемое из вершины ребро не является тем ребром, по которому мы пришли в эту вершину. Запустим dfs на нашем графе из некоторой произвольной вершины v. Так как все серые вершины лежат в стеке рекурсии, то для них вершина v достижима, так как между соседними вершинами в стеке есть ребро. Тогда, если из рассматриваемой вершины v существует ребро в серую вершину u , то это значит, что из вершины u существует путь в v и из вершины v существует путь в u состоящий из одного ребра. И так как оба эти пути не пересекаются, то цикл существует. Докажем, что цикл всегда будет найден. Пусть v- первая вершина из цикла, которая попала в рассмотрение алгортма. Тогда из v существует ребро в вершину u, которая также лежит в цикле. Так как из вершины v в вершину u существует белый путь (они лежат на одном цикле), то по лемме о белых путях во время выполнения процедуры поиска в глубину от вершины u, вершина v будет серой. Так как из u есть ребро в v, то это ребро в серую вершину. Тогда получаем,что алгоритм нащел цикл. Так как асмимптотика этого алгоритма совпадает с dfs(это по факту и есть dfs), то UCYCLE в L.

Задача 5.
Используя тот факт, что PATH, $\overline{PATH} \in$ NL, докажите, что NL = coNL.

Решение.

На семинаре было доказано, что любая задача из NL сводится к PATH (PATH $\in$ NLc), тогда если $\overline{PATH} \in$ NL, то любая задача из coNL также сводится и к $\overline{PATH}$ ($\overline{PATH} \in$ coNLc). Теперь рассмотрим произвольный язык $L \in NL$, тогда $\overline{L} \in coNL$

Задача 6.

Докажите, что язык 2SAT является NL-полным.

Решение.

Вначале покажем, что 2SAT лежит в NL . Для этого покажем, каак можно решить 2SAT при помощи PATH. Пусть ϕ-2КНФ, зависящая от переменных $x_1 \cdots x_n$. Построим по ней граф G_ϕ с 2n вершинами, которые мы будем обозначать $x_1 \cdots x_n, \overline{x_1} \cdots \overline{x_n}$, если ϕ содержит дизъюнкт $(x_i \vee x_j$, то добавим в граф ребра $\overline{x_i} \to x_j$ и $\overline{x_j} \to x_i$. Смысл заключается в следующем: если формула выполнена, но x_i ложно, то x_j должно быть истинно и наоборот. Аналогично для дизънкта $(\overline{x_i} \vee x_j)$ добавим ребра $\overline{x_j} \to \overline{x_i}$ и $x_i \to x_j$, для $(\overline{x_i} \vee \overline{x_j}$- $x_i \to \overline{x_j}$ и $x_j \to \overline{x_i}$. Тогда формула будет выполнима $\Leftrightarrow$ в построенном графе ни для одной переменной не будет одновременно путей из x_i в $\overline{x_i}$ и из $\overline{x_i}$ в x_i. Докажем это. С каждым набором переменных связаны раскраска графа в два цвета , истинный и ложный : если $x_i = 1$, то вершина x_i покрашена в истинный цвет,$\overline{x_i}$ в ложный. Иначе-наоборот. Из построения графа следуеь, что для выполняющего наобора если начало покрашено в истинный цыет, то и конец тоже, но этого не может быть, если еесть путь и из x_i в $\overline{x_i}$ и из $\overline{x_i}$ в x_i. Теперь покажем, что если для каждой вершины хотя бы одного пути нет, то выполняющий набор путей будет . Действительно, пусть нет пути из x в $\overline{x}$, тогда не может быть и двух путей из х в у и из х в $\overline{y}$. Теперь сделаем истинными х и все достижимые из него литералы. Это можно сделать непротиворечиво. Если в некотором дизъюнкте один из литералов ложен, то его отрицание достижимо из х, следовательно, достижим и второй из литералов, поэтому он истинен. Значит ни один дизъюнкт не может стать ложным. Таким образом мы свели 2SAT к проверке некоторых n путей в графе ($x_i \to \overline{x_i}$ и $\overline{x_i} \to x_i$) . Поскольку $\overline{PATH} \in NL$ это можно сделать при помощи n сертификатов, отсортированных по возрастанию i. Тогда $2SAT \in NL$. Докажем теперь полноту. Сведем $\overline{PATH}$ к 2SAT. По тройке (G,s,t) построим формулу так: для каждой вершины заведем переменную, а длоя каждого ребра $u \to v$- дизъюнкт $\overline{u} \vee v$. Кроме того, заведем отдельные дизъюнкты s и $\overline{t}$. Если пути из s в t нет, то рассмотрим набор, в котором все переменные, достижимые из s истинны, а недостижимые- ложны. Тогда любое ребро идет либо между двумя истинными переменными, либо между двумя ложными. В любом случае, $\overline{u} \vee v$ истинен, также как и дизъюнкты s и $\overline{t}$. Таким образом, формула выполнима. Обратно: пусть формула выполнима, тогда любая переменная, достижимая из s , истинна в выполняющем наборе. ЧТД