Контрольная работа 2.1

1. В задаче о максимальном разрезе задан граф G=(V, E) и требуется разбить множество вершин графа V, на два непересекающихся множества (две доли) V_1 и V_2 так, чтобы максимизировать число ребер соединяющих вершины разных множеств.
Рассмотрим следующий алгоритм локального поиска для задачи о максимальном разрезе.

- Выбрать произвольное разбиение множества вершин V.
- Если существует вершина $v \in V$ такая, что число ее соседей в доле, которой она принадлежит больше, чем число ее соседей в другой доле, то переместить эту вершину из одной доли в другую.
- Если таких вершин нет выдать полученное разбиение.

Докажите, что алгоритм остановится через конечное число шагов и решение найденное алгоритмом будет содержать по крайней мере половину всех ребер графа G.

2. Рассмотрим следующий алгоритм для задачи $P2||C_{max}$ для фиксированного $\varepsilon > 0$.

Input ($J=\{1,..., n\}$, p: $J \to \mathbf{Z}^+$)

1) Определим **Big** = $\{ j \in J| \ p_j \geq \varepsilon L\}$ и **Small** = $\{ j \in J| \ p_j < \varepsilon L\}$
2) Определим области $\boldsymbol{\Phi}^{(1)}$, $\boldsymbol{\Phi}^{(2)}$,…, $\boldsymbol{\Phi}^{(h)}$ согласно назначению больших работ по машинам.
3) Выберем область $\boldsymbol{\Phi}^{(l)}$, в которой достигается минимум величины $\max\{B_{1l},B_{2l}\}$, где B_{il} загрузка i-ой машины большими работами в назначении $\boldsymbol{\Phi}^{(l)}$.
4) Добавим маленькие работы одну за другой на машину с меньшей нагрузкой. Назовем полученное расписание σ_l.

Output (σ_l)
Показать, что полученный алгоритм является приближенной схемой.

3. Предположим, что в примере задачи о максимальной выполнимости нет одноэлементных дизъюнкций и в каждой дизъюнкции из двух элементов, по крайней мере один элемент входит без отрицания. Рассмотрим для таких примеров следующий алгоритм. Независимо для каждого i: $x_i \leftarrow 1$ с вероятностью p и $x_i \leftarrow 0$, иначе. Подберите значение параметра p , так чтобы максимизировать стоимость математического ожидания получаемых решений в худшем случае.

4. Пусть $\sigma(C_1)\circ...\circ \sigma(C_k)$ – решение, найденное алгоритмом СУПЕРСТРОКА. Доказать, что если в каждом цикле есть представитель, длина которого не превосходит вес цикла, то стоимость решения не превосходит 2OPT.