Задача 1

(a) Вершины двудольного графа можно разбить на два множества так, что каждое ребро соединяло вершины из разных множеств. Согласно определению в двудольных графах не может быть трех попарно смежных вершин, а значит это множество графов пусто $\Rightarrow$ язык принадлежит классу $\mathcal{P}$

(b) Проверим граф на связность с помощью поиска в глубину, сложность работы составляет $O(M)$, то есть алгоритм линеен по количеству ребер. Произведём серию поисков в глубину в графе: из каждой вершины, в которую мы ещё ни разу не приходили, запустим поиск в глубину, который при входе в вершину будет окрашивать её в серый цвет, а при выходе — в чёрный. Если после прохода алгоритма не все вершины окрашены, то граф несвязный, значит можно продолжать его обход, иначе — он не принадлежит языку. Для каждого связного куска проверим граф на ацикличность, это также проверяется с помощью поиска в глубину. Если в течение поиска в глубину мы пытаемся пойти в серую вершину, то это означает, что мы нашли цикл, и граф не принадлежит нашему языку
Язык принадлежит классу $\mathcal{P}$

(c)

```
FOR i, j = 1..2018
    start:
    count= 0;
    FOR k, l = 1..(n − 2018)
    IF (A[k + i, l + j] == 1)
    count + = 1
    ELSE GOTO(start)
IF count == (n-2018)*(n-2018)
    ANSWER(Принадлежит языку)
ELSE
  ANSWER(Не принадлежит языку)
```

В алгоритме 2 цикла, работающих в худшем случае по $(n - 2018)$ раз, внутри которых происходят операции $O(1)$, значит, имеем сложность алгоритма — $O(n^2)$.

Задача 2

1. $L_1, L_2 \in \mathcal{P} \Rightarrow [L_1 \bigcap L_2] \in \mathcal{P}$
Т.к. $[L_1 \bigcap L_2] \in L_1$, например, а $L_1 \in \mathcal{P}$
2. $[L_1 \bigcup L_2] \in \mathcal{P}$, т.к. $\forall \omega_1 \in L_1$ и $\forall \omega_2 \in L_2$
выполняется $\omega_{1,2} \in \mathcal{P}$
3. Доказать, что $L_1 L_2 \in P$, если $L_{1,2} \in P$
Напомним, что $L = L_1 L_2 = \{\exists \omega_1 \in L_1; \omega_2 \in L_2 : \forall \omega \in L \longmapsto \omega = \omega_1 \omega_2\}$
4. По условию языки $L_{1,2}$ принимаются за полиномиальное время $O(n^k)$ (k — константа)
т.е. любая n-символьная строка x из языков $L_{1,2}$ принимается за время $O(n^k)$
Рассмотрим произвольное слово ω языка L. Причём существуют такие $\omega_{1,2}$, что $\omega = \omega_1 \cdot \omega_2$
$\omega_1 \in L_1$, $\omega_2 \in L_2$. Известно, что слова $\omega_{1,2}$ принимаются за полиномиальное время $\Rightarrow$ всё слово ω принимается за *полиномильное время.*
5. $L \in \mathcal{P} \Rightarrow$ существует класс предикатов $\mathcal{P}$, который определяет язык L (т.е. $\forall \omega \in L \longmapsto P(\omega) = 1; \forall \omega \notin L \longmapsto P(\omega) = 0$). Тогда $\forall \omega \in \overline{L} \longmapsto P(\omega) = 0; \forall \omega \notin \overline{L} \longmapsto P(\omega) = 1$. Получается, что тот же класс предикатов определил язык $\overline{L}$ (существует конечное число наборов значений, при которых предикат обращается в ноль, то есть существует определенный набор *слов*, на которых предикат принимает определённое значение (в нашем случае 0), а на всех других (на словах, которые не входят в язык $\overline{L}$) предикат принимает *другое* значение (в нашем случае 1) $\Rightarrow$ язык $\overline{L}$ задан.
6. $L \in \mathcal{P}$
По определению замыкания Клини $L^* = \epsilon + L + L^2 + ... + L^k + ...$
L^k - конкатенация (лежит в классе $\mathcal{P}$)
$L^i + L^j$ - объединение (лежит в классе $\mathcal{P}$)
ϵ - лежит в классе $\mathcal{P}$
$\Downarrow$
$L^* \in \mathcal{P}$

Задача 3

Доказать: $P \subseteq co - NP$
Воспользуемся: $P \subseteq NP$, т.е. $\forall L \in P \to L \in NP \Rightarrow \overline{L} \in co - NP \Rightarrow P \subseteq co - NP$

Задача 4

Доказать, что $NP \neq co - NP \Rightarrow NP \neq P$
Доказательство приведём от противного, т.е. предположим, что если $NP \neq co - NP \Rightarrow NP = P$
Тогда для любого языка $L \in P \ \Rightarrow L \in NP \Rightarrow \overline{L} \in co - NP \Rightarrow co - NP = NP \Rightarrow$ противоречие! $\Rightarrow$ наше предположение неверно $\Rightarrow NP \neq co - NP$

Задача 5

Необходимо доказать: $L \leq_p \overline{L} \Leftrightarrow \overline{L} \leq_p L$
Заметим, что $\overline{L} \cap L$ — пустое множество
$\forall x \in \Sigma^* x \in L \Leftrightarrow f(x) \notin L \Longrightarrow f(x) \in \overline{L}$
$\forall x \in \Sigma^* x \notin L \Longrightarrow \forall x \in \Sigma^* x \in \overline{L} \Leftrightarrow f(x) \in L$
Где $f(x)$ вычислимая за $O(n^k)$ функция
$\Downarrow$ (по определению полиномиальной сходимости)
$L \leq_p \overline{L} \Leftrightarrow \overline{L} \leq_p L$