Задача 1

$x = 41;\ y = x^e \ mod\ N = 41^3\ mod\ 391 = 105$
В открытом доступе $y = 105,\ N = 391,\ e = 3.$
Зная p и q, получаю $d = e^{-1}\ mod\ (p-1)(q-1) = 3^{-1}\ mod\ 352 = 235$
Затем дешифрую $x = y^d\ mod\ N = 105^{235}\ mod\ 391 = 41$

Задача 2

Злоумышленнику известны N, e, d. Он находит $p, q : N = pq$. Нахождение обратного к 3 по модулю $M = (p-1)(q-1)$ из уравнения $3d + Mb = 1$, где $b \in Z$, возможно по двум схемам в зависимости от остатка M при делении на 3:

1)Если остаток при делении M на 3 равен 1:

$$3d + Mb = 1$$

$$3\begin{pmatrix}1\\0\end{pmatrix} \qquad M\begin{pmatrix}0\\1\end{pmatrix}$$

$$3\begin{pmatrix}1\\0\end{pmatrix} \qquad 1\begin{pmatrix}-\left\lfloor\frac{M}{3}\right\rfloor\\1\end{pmatrix}$$

$d = -\left\lfloor\frac{M}{3}\right\rfloor, b = 1; \quad M = -3d + 1$

2)Если остаток при делении M на 3 равен 2:

$$3d + Mb = 1$$

$$3\begin{pmatrix}1\\0\end{pmatrix} \qquad M\begin{pmatrix}0\\1\end{pmatrix}$$

$$3\begin{pmatrix}1\\0\end{pmatrix} \qquad 2\begin{pmatrix}-\left\lfloor\frac{M}{3}\right\rfloor\\1\end{pmatrix}$$

$$1\begin{pmatrix}1+\left\lfloor\frac{M}{3}\right\rfloor\\-1\end{pmatrix} \qquad 2\begin{pmatrix}-\left\lfloor\frac{M}{3}\right\rfloor\\1\end{pmatrix}$$

$d = 1 + \left\lfloor\frac{M}{3}\right\rfloor, b = -1; \quad M = 3d - 1.$

Поскольку $N - M + 1 = p + q = s$ (обозначим так) и $N = pq$, то получаем: $p = s - q$ и $N = (s-q)q \Rightarrow q^2 - sq + N = 0$. Отсюда $q = 0.5(s + \sqrt{s^2 - 4N})$. Для вычисления M есть два варианта, но правильным будет только один, поскольку M не может иметь два разных остатка. Поэтому посчитаем для обоих M p и q. В одном из вариантов получатся простые числа, что и будет ответом.

Задача 3

$N = 2021, e = 25, M = (p-1)(q-1) = 42 \cdot 46 = 1932, p = 43, q = 47$: Найдем обратный к e по модулю M.

$$25d + 1932b = 1$$

$$25\begin{pmatrix}1\\0\end{pmatrix} \qquad 1932\begin{pmatrix}0\\1\end{pmatrix}$$

$$25\begin{pmatrix}1\\0\end{pmatrix} \qquad 7\begin{pmatrix}-77\\1\end{pmatrix}$$

$$4\begin{pmatrix}232\\-3\end{pmatrix} \qquad 7\begin{pmatrix}-77\\1\end{pmatrix}$$

$$4\begin{pmatrix}232\\-3\end{pmatrix} \qquad 3\begin{pmatrix}-309\\4\end{pmatrix}$$

$$1\begin{pmatrix}541\\-7\end{pmatrix} \qquad 3\begin{pmatrix}-309\\4\end{pmatrix}$$

Получаем $d = 541$ — степень, в которую надо возвести сообщение для электронной подписи.

Задача 4

а)Используем шаги бинарного поиска с одним сравнением выбранного элемента с элементом правее него на каждом шаге. На каждом шаге выбираем половину, в направлении которой значение растет (если элементы равны, то выбор произвольный). В этой половине действительно будет горка, поскольку, если значения продолжат расти, то горка будет в конце, если начнут убывать, то горка будет внутри отрезка, как и если значение постоянно с некоторого момента до конца отрезка. Алгоритм закончится, поскольку длина рассматриваемых отрезков убывает вдвое на каждом шаге. Шагов будет $\lfloor \log_2 n \rfloor$, на каждом из которых ровно одно сравнение.

б)В любом массиве есть горка. Начиная просмотр с каждого конца массива получаем, что горки нет, если значения сторого возрастают при движении к середине массива. Но тогда в середине обязательно будет горка. Поэтому для любого массива движение в сторону увеличения значения элементов приведет нас к горке. Попытаемся как можно быстрее двигаться в сторону роста значений массива. Для этого будем уменьшать длину рассматриваемого участка значений, где точно есть горка. Если соотношение частей при выборе элемента не равно $1:1$, то в худшем случае будет выбираться большая часть, что приведет к большему числу сравнений. При выборе среднего элемента получаем минимальное число сравнений, поскольку при этом длина рассматриваемого отрезка убывает быстрее всего.

Задача 5

1)Клика переходит в независимое множество в реберном дополнении графа.

2)Наличие двух клик на дизьюнктых подмножествах вершин в графе означает наличие двух долей в реберном дополнении графа.

3)Двудольность реберного дополнения графа равносильна условию задачи. Проверка двудольности производится за полином времени: начинаем BFS из произвольной вершины, окрасив ее в цвет 1. Всех ее соседей красим в цвет 0 и так далее, чтобы цвета соседей были разные. Если в процессе обхода найдется пара соседних вершин с одинаковыми цветами, то ответ «Нет», иначе — «Да».

Значит, данный язык не является NPc.

Задача 6

Решим данную задачу за полином времени. Возвращение и повторный проход по ребрам увеличивает длину обычного пути на четное число. Если есть пути длины 10 и 11, то обратными проходами можно увеличить их длину до любого $S \geq 10$. Проверим наличие путей длины 10 и 11 между s и t возведением матрицы смежностей в эти степени. В итоге, задача решается за $O(|V|^3)$. Значит, она принадлежит co-NP, поскольку принадлежит P.

Задача 7

Отсортируем ребра по убыванию их веса. Далее будем добавлять вершины ребер в множества двух цветов (в лучшем случае это две доли) , начиная с ребер большего веса, так, что их вершины имеют по возможности разные цвета. Делаем это, пока каждая из вершин не попадет в какое-то множество.
При этом получаем оптимальную для задачи раскраску графа, поскольку при таком алгоритме ребра с большими весами исключаются из рассмотрения наилучшим образом, и вследствие этого, максимальный вес ребра с одноцветными концами получается минимальным по всем возможным раскраскам.
Сортировка работает за $O(m \log m)$.