1.

а) При решении уравнения вида $ax + by = c$ возможны две ситуации, когда c делится на НОД(a, b), тогда уравнение имеет решения в целых числах и когда нет – тогда решений нет: при любых целых x и y число $ax + by$ делится на НОД(a, b) и поэтому не может равняться числу c, которое на НОД(a, b) не делится.

Алгоритмом Евклида найдем НОД(a, b):

НОД$(238, 385)$ = НОД$(385, 238)$ = НОД$(238, 147)$ = НОД$(147, 91)$ = НОД$(91, 56)$ = НОД$(56, 35)$ = НОД$(35, 21)$ = НОД$(21, 14)$ = НОД$(14, 7)$ = НОД$(7, 0)$ = 7.

Таким образом, НОД$(a, b) = 7$. $\frac{133}{7} = 19$. Следовательно, уравнение имеет решения в целых числах.

Разделим обе части уравнения на НОД$(a, b) = 7$:

$34x + 55y = 19$

Необходимо найти все решения данного уравнения, заметим, что, если $x^{'}$ и $y^{'}$ – решение, тогда и $x = x^{'} + bt$ и $y = y^{'} - at$ – решение, где $t \in \mathbb{Z}$:

$a(x^{'} + bt) + b(y^{'} - at) = ax^{'} + by^{'} = c$

Значит, необходимо найти для начала любое частное решение уравнения.

Сделаем замену: $x = 19x_0, y = 19y_0$, тогда:

$34x_0 + 55y_0 = 1$

Будем решать данное уравнение расширенным методом Евклида.

$55x_1 + 34y_1 = 1$

$34x_2 + 21y_2 = 1$

$21x_3 + 13y_3 = 1$

$13x_4 + 8y_4 = 1$

$8x_5 + 5y_5 = 1$

$5x_6 + 3y_6 = 1$

$3x_7 + 2y_7 = 1$

$2x_8 + y_8 = 1$

$x_9 = 1 \; y_9 = 0$

Теперь нам необходимо подняться наверх и восстановить x_0 и y_0.

$ax + by = c$

$y = \frac{c-ax}{b} = c^{'} - a'x$

$c^{'} - a'x = bt$

$a^{'}x + bt = c'$

То есть, сделав замену: $b_1y_1 + a_1x_1 = c_1$, где $a_1 = b, b_1 = a \, mod \, b$

Таким образом, получим в общем виде:

$$\begin{cases} y_n = \frac{1-a_ny_n+1}{b_n} \\ x_n = y_{n+1} \end{cases}$$

Тогда получим:

$y_8 = \frac{1-2*0}{1} = 1, x_8 = 0$

$y_7 = -1, x_7 = 1$

$y_6 = 2, x_6 = -1$

$y_5 = -3, x_5 = 2$

$y_4 = 5, x_4 = -3$

$y_3 = -8, x_3 = 5$

$y_2 = 13, x_2 = -8$
$y_1 = -21, x_1 = 13$
$y_0 = 13, x_0 = -21$

Таким образом, общее решение, согласно формулам, написанным выше, получится: $x = 19 * (-21) + 55t, y = 19 * 13 - 34t$, то есть, $\mathbf{x = -399 + 55t, y = 247 - 34t}$

б) $143x+121y = 52$ Найдем НОД$(143, 121)$ = НОД$(121, 22)$ = НОД$(22, 11)$ = НОД$(11, 0)$ = 11

Разделим обе части уравнения на НОД$(a, b) = 11$:

Однако 52 не делится на 11, это уравнение попадает под второй случай, описанный выше. Следовательно, решений нет.

2.

Построим цепочку:

$7^{13}\, mod\, 167 = ((7^6)^2 * 7)\, mod\, 167 = 81^2 * 7\, mod\, 167 = 2$
$\uparrow$
$7^6\, mod\, 167 = (7^3)^2\, mod\, 167 = 81\, mod\, 167 = 81$
$\uparrow$
$7^3\, mod\, 167 = (7^2 * 7)\, mod\, 167 = 49 * 7\, mod\, 167 = 9$
$\uparrow$
$7^2\, mod\, 167 = (7 * 7)\, mod\, 167 = 49$
$\uparrow$
$7\, mod\, 167 = 7$

Таким образом, ответ: $\mathbf{7^{13}\, mod\, 167 = 2}$

3.

Для начала немного подробнее распишем алгоритм:
$x = qy + r$. Ищем такие q и r.
Если x – четно: $\frac{x}{2} = qy + r, x = 2qy + 2r$
Если x – нечетно: $\frac{x-1}{2} = qy + r, x = 2qy + 2r + 1$
Если r стало больше y в ходе умножения на 2, то $r = r - y, q = q + 1$

Корректность:

Докажем корректность алгоритма по методу математической индукции.

0. Для $\forall y \geq 1, x = 0$ алгоритм работает верно

1. Для $\forall y \geq 1, \forall i < x$ алгоритм работатет верно по предположению, то есть, пара чисел $(q, r) = Divide([x/2], y)$ посчитана верно.

2. При делении x на 2 и округлении вниз мы просто отрезаем последний бит в двоичной записи числа, затем при умножении на 2 мы сдвигаем на один бит влево. Теперь все зависит от последнего бита x: если он равен 0, то все, если нет – то надо к r прибавить 1 и если r стало больше y в ходе умножения на 2, то $r = r - y, q = q + 1$ (корректность этих математичесских формул описана выше).

Сложность:

На вход поступают два числа x и y. Длина входа: $log\, x + log\, y = O(n)$, n – число битов.

Если построить стек рекурсии, то вызовов будет $log\, x$, так как на каждом шаге аргумент x делится пополам и возможное прибавление 1 не играет роли, так как x округляется вниз. Таким образом, глубина рекурсии – $log\, x = O(n)$

На каждом шаге рекурсии выполняется деление и умножение на 2 x и умножение на 2 y– это побитовый сдвиг влево ($O(1)$). Операции сложения в худшем случае, наверху рекурсии, стоят $O(n)$. (оценка $log\, x$ и $log\, y$). Таким образом, на каждом шаге рекурсии, глубина которой $O(n)$, выполняются операции, которые стоят $O(n)$. Следовательно, итоговая сложность – $\mathbf{O(n^2)}$.

4.

1. F(3,5):
$5 = 101 \rightarrow XSSX$
$a = XSSX, y = 1$
$a[1] == X : y = 1 * 3 = 3$
$a[2] \neq X : y = 3 * 3 = 9$
$a[3] \neq X : y = 9 * 9 = 81$
$a[4] == X : y = 81 * 3 = 243$
Следовательно, $F(3,5) = 243$
2. $F(x,m) = x^m$ – возведение x в степень m.
3. Данный алгоритм – подобие алгоритма быстрого возведения в степень для числа в двоичной записи.

Для начала кратко опишем этот алгоритм.

При переводе степени m в двоичный вид, мы имеем запись вида:
$\overline{n_k n_{k-1} ... n_0}$, где $n_i = \{0, 1\}$

Если $n_i = 1$, то текущий результат возводится в квадрат и затем умножается на x (число, которое мы возводим в степень). Если $n_i = 0$, то текущий результат просто возводится в квадрат. То есть, если степень четна – возводим в квадрат, если нет – возводится в квадрат и затем умножается на x. Данный алгоритм был реализован на примере во втором номере.

Заметим, что предложенный алгоритм делает похожие операции: просто 1 закодирована двумя символами, каждый из которых выполняет одно из действий: возведение в квадрат, умножение на x, то есть, в итоге получаем тот же результат. Для нуля, который закодирован одной буквой S: выполняется именно возведение в квадрат.

В начале вычеркивается символ S, как незначащий 0 в двоичной записи числа, который появляется, если первая цифра в двоичной записи – 1. Следовательно, данный алгоритм возводит число в степень, но в отличие от классического алгоритма для чисел в двоичном представлении, делает это для чисел в иной кодировке.

4. Пусть k – длина показателя m в битах: $k = log\, m$. На каждом шаге по закодированной последовательности a совершается одна операция, которая по условию стоит $O(1)$. Закодированная последовательность содержит не более, чем $2k-1$ символов. Следовательно, сложность алгоритма: $O(log\, m)$

5.

1. $T_1(n) = T_1(n-1) + cn$
$T_1(n-1) = T_1(n-2) + c(n-1)$
То есть, $T_1(n) = T_1(n-2) + c(n-1) + cn$ и т.д.

Таким образом, заметим, что $T_1(n) = T_1(3) + 4c + 5c + ... + c(n-1) + cn = 1 + c(4 + 5 + ... + n) = 1 + c(\frac{n(n+1)}{2} - 1 - 2 - 3) = \mathbf{\Theta(n^2)}$

2. $T_2(n) = T_2(n-1) + 4T_2(n-3)$

Решим данное рекуррентное уравнение через характеристический многочлен.

$x^3 = x^2 + 4$

$x^3 - x^2 - 4 = 0$

$x_1 = 2, x_2 = \frac{1}{2}(-1 + i\sqrt{7}), x_3 = \frac{1}{2}(1 + i\sqrt{7})$

То есть общее решение будет:

$T_2(n) = A(2)^n + B(1.4)^n cos(110.7n) + C(1.4)^n sin(110.7n)$

Следовательно, $T_2(n) = \Theta(2^n)$, то есть, $\mathbf{\log T_2(n) = \Theta(n)}$

3. Из второго пункта $\mathbf{T_2(n) = \Theta(2^n)}$.