Пункт 1.

- $T(n) = 7 \cdot T(n/3) + n^2$,
- $T(n) = 4 \cdot T(n/2) + \log n$,
- $T(n) = 0{,}5 \cdot T(n/2) + 1/n$,
- $T(n) = 3 \cdot T(n/3) + n/2$.

1. $T(n) = 7 * T(\frac{n}{3}) + n^2$
В данном случае $a = 7 \geq 1, b = 3 > 1$. a и b не зависят от n.
Дополнительная работа на каждом шаге рекурсии $f(n)$ представима в виде $f(n) = \Theta(n^c),\ \ c = 2$. c не зависит от n, оценкой временной сложности $f(n)$ является мономом $\implies$ можно применить master-теорему.

$\log_b a = \log_3 7 < \log_3 9 = 2 = c \implies c > \log_b a$
$(c > \log_b a) \wedge (f(n) = \Theta(n^c)) \implies$ по второй формулировке master-теоремы $T(n) = \Theta(n^c) = \Theta(n^2)$

Ответ: $T(n) = \Theta(n^2)$

2. $T(n) = 4 * T(\frac{n}{2}) + \log n$
В данном случае $a = 4 \geq 1, b = 2 > 1$. a и b не зависят от n.
Дополнительная работа на каждом шаге рекурсии $f(n)$ представима в виде $f(n) = \Theta(\log n)$

$f(n) = \Theta(\log n) \implies f(n) = \mathcal{O}(\log n) \implies \forall k > 0 : f(n) = \mathcal{O}(n^k)$
Положим $k := 1 = 2 - 1 = \log_2 4 - 1 = \log_b a - 1 \implies$ по первой формулировке master-теоремы при $\epsilon = 1$ $\ f(n) = \mathcal{O}(n^{\log_b a - \epsilon}) \implies T(n) = \Theta(n^{\log_b a}) = \Theta(n^2)$

Ответ: $T(n) = \Theta(n^2)$

3. $T(n) = 0.5 * T(\frac{n}{2}) + \frac{1}{n}$
В данном случае $a = 0.5 < 1 \implies$ применение master-теоремы невозможно.

Ответ: применение master-теоремы невозможно, т.к. $a = 0.5 < 1$

4. $T(n) = 3 * T(\frac{n}{3}) + \frac{n}{2}$
В данном случае $a = 3 \geq 1, b = 3 > 1$. a и b не зависят от n.
Дополнительная работа на каждом шаге рекурсии $f(n)$ представима в виде $f(n) = \Theta(n^c),\ \ c = 1$. c не зависит от n, оценкой временной сложности $f(n)$ является мономом $\implies$ можно применить master-теорему.

$\log_a b = \log_3 3 = 1 = c \implies f(n) = \Theta(n^{\log_b a}) \implies$ по первой формулировке master-теоремы $T(n) = \Theta(n^{\log_b a} \log n) = \Theta(n \log n)$

Ответ: $T(n) = \Theta(n \log n)$

Пункт 2.

Master-теорема не применима для оценки функции временной сложности $T(n) = 0.5 * T(\frac{n}{2}) + \frac{1}{n}$

При помощи метода подстановки докажем, что $\mathcal{O}(\log n)$ является возможной асимптотически верхней границей данной функции временной сложности.

$T(n) = \mathcal{O}(\log n) \iff (\exists n_0 \in \mathbb{N}\ \exists c > 0\ \forall n \geq n_0 \implies T(n) \leq c \log_2 n)$

Покажем, что при $T(n) = \mathcal{O}(\log n)$ $\exists n_0 \in \mathbb{N}\ \exists c > 0\ \forall n \geq n_0 \implies 0.5 * T\left(\frac{n}{2}\right) + \frac{1}{n} \leq c \log_2 n$

$0.5 * T\left(\frac{n}{2}\right) + \frac{1}{n} \leq 0.5 * c \log_2 \frac{n}{2} + \frac{1}{n} \leq 0.5 * c \log_2 \frac{n}{2} + 1$ при $n \geq 1$

$0.5 * c \log_2 \frac{n}{2} + 1 = 0.5 * c \log_2 n - 0.5c + 1 = 0.5 * c \log_2 n$ при $c = 2$

При $c > 0 \wedge n \in \mathbb{N} \implies 0.5 * c \log_2 n \leq c \log_2 n$

Получили: $(n \geq 1 \wedge c = 2 \implies 0.5 * T\left(\frac{n}{2}\right) + \frac{1}{n} \leq c \log_2 n) \implies$

$\exists n_0 = 1 \in \mathbb{N}\ \exists c = 2 > 0\ \forall n \geq n_0 \implies T(n) \leq c \log_2 n$

Следовательно, $f(n) = \mathcal{O}(\log n)$

$q.e.d.$

Ответ: $T(n) = 0.5 * T\left(\frac{n}{2}\right) + \frac{1}{n} = \mathcal{O}(\log n)$