Traccia della soluzione dell'esercizio 1.
Applichiamo una BFS sull'albero. La BFS memorizza su ogni nodo $v_i \in N$ la sua distanza $d(v_i)$ dalla radice e costa $\Theta(N + A)$ che su un albero è $\Theta(N)$ visto che $A = \Theta(N)$. Sia V un vettore che memorizza in posizione i il numero di nodi distanti i dalla radice dell'albero. Per costruire V basta sorrere tutti i nodi v_i e eseguire l'istruzione $V(d(v_i)) = V(d(v_i)) + 1$ (costo $\Theta(N)$). A questo punto basta calcolare il massimo di V (costo $\Theta(N)$). Costo totale $\Theta(N)$.

Traccia della soluzione dell'esercizio 2.
Prima soluzione (scandalosa 2 punti): Copiamo X e Y dentro Z e lo ordiniamo (costo $\Theta(n\ log(n))$) e stampiamo $Z(n)$.
Seconda soluzione (accettabile almeno 7 punti): Copiamo X e Y dentro Z e stampiamo $Selection(Z, n)$ (costo $\Theta(n)$)
Terza soluzione (almeno 7 punti): Facciamo un $Merge$ di X e Y visto che sono già ordinati (costo $\Theta(n)$) Otteniamo un vettore Z ordinato e stampiamo $Z(n)$.
Quarta soluzione (ottima 13 punti): Gli elementi di X e Y sono tutti distinti. Sia $n = 2m + 1$. $M_X = X(m + 1) = mediano\ di\ X$ e $M_Y = Y(m + 1) = mediano\ di\ Y$. Assumiamo senza perdere di generalità che $M_X < M_Y$. Supponiamo che il vettore Z sia l'unione di X e Y e sia anche ordinato (non lo ordiniamo, ma immaginiamocelo ordinato). All'interno di Z, M_X può stare nelle posizioni comprese tra $m + 1$ e $2m + 1$, mentre M_Y può stare nelle posizioni comprese tra $2m + 2$ e $3m + 2$. M_Z è in posizione $n = 2m + 1$ quindi non può essere uguale a M_Y, ma può essere uguale a M_X. Di sicuro $M_X \leq M_Z < M_Y$.
Idea algoritmo ottimo:
$Y(m)$ è l'elemento in Y subito a sinistra di M_Y. Se $M_X > Y(m)$ allora M_Z è uguale a M_X. Fine. Se $M_X < Y(m)$ allora buttiamo via la prima metà di X e la seconda metà di Y e riapplichiamo il metodo ricorsivamente su $X' =$ prima metà di X e $Y' =$ seconda metà di Y cercando l'elemento in $Z' = X' \cup Y'$ in posizione $m + 1$ e non più $2m + 1$. In pratica ogni volta dimezziamo la dimensione del problema con un costo costante.

Traccia della soluzione dell'esercizio 3.
$T(n) \geq n^2$ semplicemente perché $T(n) = 7\ T(\frac{n}{4}) + 2\ T(\frac{n}{8}) + n^2 \geq n^2$.
Supponiamo adesso che $\forall m < n$ valga $T(m) \leq c\ m^2$, abbiamo
$T(n) = 7\ T(\frac{n}{4}) + 2\ T(\frac{n}{8}) + n^2$ che per ipotesi induttiva è minore o uguale di

$7c\ (\frac{n}{4})^2 + 2c\ (\frac{n}{8})^2 + n^2$ che è minore di cn^2 per $c \geq 32/17$.

Quindi $T(n) = \Theta(n^2)$.