(a) Il linguaggio generato dalla grammatica G è costituito da tutte le stringhe di simboli a e b che possono essere ottenute in modo ricorsivo applicando le regole di produzione della grammatica G. In particolare, la grammatica genera stringhe che iniziano e terminano con lo stesso simbolo, e tali stringhe possono contenere una qualsiasi combinazione di simboli a e b tra i due simboli di inizio e fine. Quindi, possiamo descrivere il linguaggio generato come:

$L = \{w \mid w \in \{a, b\}^*$ e w ha la forma $a^n x a^n$ o $b^n x b^n$ per qualche $x \in \{a, b\}^*$ e $n \geq 0\}$

In altre parole, il linguaggio L contiene tutte le stringhe di simboli a e b che possono essere suddivise in tre parti: una parte iniziale formata da un numero qualsiasi di simboli a o b, una parte centrale formata da una combinazione qualsiasi di simboli a e b, e una parte finale che è una copia della parte iniziale scritta al contrario.

(b) Per dimostrare che $L = L(G)$, dobbiamo dimostrare che ogni stringa w generata dalla grammatica G appartiene a L, e viceversa, che ogni stringa in L può essere generata dalla grammatica G.

Per dimostrare la prima direzione, ovvero che ogni stringa w generata da G appartiene a L, possiamo dimostrare per induzione sulla lunghezza di w che w ha la forma $a^n x a^n$ o $b^n x b^n$ per qualche $x \in \{a, b\}^*$. Se la lunghezza di w è 0, allora $w = \varepsilon$ e quindi appartiene a L. Altrimenti, consideriamo le regole di produzione utilizzate per generare la prima e l'ultima parte di w. Se la regola utilizzata è $S \to \varepsilon$ o $S \to a$ o $S \to b$, allora la parte iniziale e finale di w sono vuote o consistono di un singolo simbolo. In questo caso, possiamo considerare la parte centrale di w come la parte iniziale e finale scritte al contrario, e quindi w è nella forma richiesta. Altrimenti, la regola utilizzata è $S \to aSa$ o $S \to bSb$. In questo caso, la parte centrale di w ha la forma $a^n-1 x a^n-1$ o $b^n-1 x b^n-1$ per qualche $x \in \{a, b\}^*$. Per ipotesi induttiva, possiamo assumere che la parte centrale sia nella forma richiesta, e quindi w ha la forma $a^n x a^n$ o $b^n x b^n$ come richiesto.

Per dimostrare la seconda direzione, ovvero che ogni stringa in L può essere generata dalla grammatica G, possiamo costruire una derivazione per ogni stringa in L. Consideriamo una stringa $w \in L$ di lunghezza n. Se $n = 0$, allora $w = \varepsilon$ e la derivazione consiste solo della regola $S \to \varepsilon$. Altrimenti, consideriamo la suddivisione di w in tre parti come descritto nella descrizione del linguaggio L. La parte iniziale e finale di w hanno la forma a^n o b^n, e possono essere generate utilizzando le regole $S \to aSb$ o $S \to bSa$. La parte centrale di w può essere generata utilizzando la regola $S \to aSa$, bSb o bSa, aSb. Quindi, possiamo generare la stringa w applicando le regole di produzione della grammatica G in modo opportuno.

(c) Il linguaggio L è un linguaggio context-free, ma non è un linguaggio regolare. La dimostrazione che L è context-free segue dalla dimostrazione che $L = L(G)$ nella parte (b). Infatti, la grammatica G è libera dal contesto e genera il linguaggio L, quindi L è un linguaggio context-free.

Per dimostrare che L non è un linguaggio regolare, possiamo utilizzare il pumping lemma per i linguaggi regolari. Supponiamo per assurdo che L sia regolare e sia generato da un automa a stati finiti M con n stati. Consideriamo la stringa $w = a^n b^{n^3} \in L$, dove n è il parametro del lemma. Poiché w appartiene a L, esiste una suddivisione di w nella forma $w = xyz$, dove:

1. $|xy| \leq n$2. $|y| > 0$

3. $xy^iz \in L$ per ogni $i \geq 0$

Consideriamo la suddivisione di w in modo che la sottostringa y contenga solo simboli a, ovvero $y = a^k$ per qualche $k > 0$. Allora, la parte xz di w contiene esattamente $n + n^3 - k$ simboli a e n^3 simboli b. Poiché $|xy| \leq n$, la sottostringa xy deve contenere solo simboli a, ovvero $xy = a^j$ per qualche $j \leq k$.

In particolare, se scegliamo $i = 2$, allora xy^2z contiene $k + j$ simboli a e n^3 simboli b. Dato che $k > 0$, abbiamo $k + j < n + n^3$, e quindi $k + j < n^3$. Quindi, se L fosse regolare, xy^2z dovrebbe ancora appartenere a L, il che implica che k + j deve essere uguale a n^3. Ma questo è impossibile, poiché abbiamo dimostrato che k + j < n^3, e quindi abbiamo ottenuto una contraddizione.

Concludiamo quindi che L non può essere generato da un automa a stati finiti, e quindi non è un linguaggio regolare.