Solution.

(a) Let i be the current value of our algorithm at some recursive step. In the case where the algorithm does not stop, we set $i \to i - F_k$, where F_k is the maximum Fibonacci term such that $F_k \leq i$. Consider now if F_{k-1} was the largest Fibonacci number that fits in $n - F_k$. This would imply

$$\begin{aligned} n - F_k - F_{k-1} &\geq 0 \\ n - (F_k + F_{k-1}) &\geq 0 \\ n - F_{k+1} &\geq 0 \\ n &\geq F_{k+1}. \end{aligned}$$

This, however, is a contradiction as it would imply that F_{k+1} fits into n, and so F_k is not the largest. Thus, the next highest must be at most F_{k-2}. Thus, our sequence would be a sum of non-consecutive Fibonacci numbers. Each iteration will never reach a negative point, but it will always be decreasing. Once we reach $i = 0$, 1 or 2, the algorithm stops. Thus, this algorithm works.

(b) Any sum of distinct, non-consecutive Fibonacci numbers that are at most F_i is less than F_{i+1}. This can be proven by induction as follows.

- *Base case*: For $i = 2$, it's clear that the only set to consider would be $\{F_2\} = \{F_1\}$, who's sum is $1 < 2 = F_3$.
- *Inductive step*: Assume our statement holds for all sets of distinct, non-consecutive Fibonacci numbers whose largest member is F_i. That is, the sum of all members in this set is less than F_{i+1}. Consider now any non-empty set S of distinct, non-consecutive Fibonacci numbers whose largest member is F_{i+1}. Let T be a set of distinct, non-consecutive Fibonacci numbers with a largest member F_{i-1} (since the members are non-consecutive) such that $S = \{F_{i+1}\} \cup T$. Therefore, by our inductive hypothesis,

$$\sum_{s \in S} s = F_{i+1} + \sum_{t \in T} t < F_{i+1} + F_i = F_{i+2}.$$

By the principle of mathematical induction, our statement has been proven true. Now, consider two non-empty sets S_1 and S_2 of distinct, non-consecutive Fibonacci numbers which have the same total sum. If we remove any of the elements in common from these sets, the resulting sets must be equal in terms of their sums as well - because we are removing equal values from both sets. Assume that these new sets, say S_1' and S_2', are non-empty. The largest terms in each of them must then not be equal since $S_1' \cap S_2' = \varnothing$. Without loss of generality, assume that $F_{s_1} < F_{s_2}$, where F_{s_1} and F_{s_2} are the largest terms in S_1' and S_2' respectively. It must be the case that, by our induction proof,

$$\sum_{x \in S_1'} x < F_{s_1+1} \leq F_{s_2}.$$

But we know that $\displaystyle\sum_{x \in S_2'} x \geq F_{s_2}$, thus we have a contradiction and so $S_1' = S_2' = \varnothing$.

Ultimately, we have proven that S_1 and S_2 cannot have any differences, so the algorithm from part (a) must produce a unique output.