1a. Let's call the number of solutions for n seating positions arranged in a straight line as $S(n)$. In any such solution, either the nth seat is chosen or not. The number of solutions for the latter case is $S(n-1)$, and that for the former case is $S(n-2)$ (because when the nth seat is chosen, the $(n-1)$th seat cannot be). Thus, $S(n) = S(n-1) + S(n-2)$. For the base case, we can easily see that $S(0) = 1$ (because the act of not choosing anything is itself a solution) and $S(1) = 2$ (because we can either choose the only seat or not). Thus, we see that $S(n)$ has the same recurrence as $F(n)$, the Fibonacci sequence, but $S(0) = F(1)$ and $S(1) = F(2)$. Therefore, by induction, $S(n) = F(n+1)$. This concludes the proof.

1b. Let's call the number of solutions for n seating positions arranged in a circle as $T(n)$. We can see that almost any solution the problem where seats are arranged in a straight line can be a solution to the case where the seats are arranged in a circle. The only cases we must exclude are those where seat 1 and seat n are both chosen. For $n >= 4$, the number of such cases is $S(n-4)$, since both 1 and n are chosen, neither seat 2 nor seat $n-1$ can be. So, the number of solutions for n seats arranged in a circle is simply $S(n) - S(n-4) = F(n+1) - F(n-3) = F(n) + F(n-1) - F(n-3) = F(n) + F(n-2)$. For the cases $n = 3$ and $n = 2$, we can manually verify that this equation holds. This concludes the proof.

2a. Consider the two expressions $F_n^2 - F_{n+1}F_{n-1}$ and one of similar form obtained by replacing n with $n-1$, $F_{n-1}^2 - F_nF_{n-2}$. We will first prove that these two terms add to 0.

$$
\begin{array}{ll}
F_n^2 - F_{n+1}F_{n-1} + F_{n-1}^2 & -F_nF_{n-2} \\
F_n^2 - F_{n-1}(F_{n+1} - F_{n-1}) & -F_nF_{n-2} \\
F_n^2 - F_{n-1}F_n & -F_nF_{n-2} \\
\quad F_n(F_n - F_{n-1}) & -F_nF_{n-2} \\
\quad F_nF_{n-2} & -F_nF_{n-2} \\
\ 0 &
\end{array}
$$

Thus, such expressions have the same magnitude but alternate signs each step along the way (from $n = 0$ onwards). We can check the base case value of $F_1^2 - F_2F_0 = -1$ to infer that $F_n^2 - F_{n+1}F_{n-1} = (-1)^n$. This concludes the proof.

2b. This one can be derived by repeated application of the recursive definition of the Fibonacci series.

$$
\begin{aligned}
F_{n+2} &= F_n + F_{n+1} \\
F_{n+2} &= F_n + F_{n-1} + F_n \\
F_{n+2} &= F_n + F_{n-1} + F_{n-2} + F_{n-1} \\
&\dots \\
F_{n+2} &= \sum_{i=n}^{0} F_i + F_1
\end{aligned}
$$

This concludes the proof.

2c. We can easily see that $F_2 = F_0^2 + F_1^2 = 2$ and $F_3 = F_1(F_0 + F_2) = 3$. Assume that the given equations are true for up to $2n+1$. Then, we can derive the following for F_{2n+2} and F_{2n+3}.

$$
\begin{aligned}
F_{2n+2} &= F_{2n} + F_{2n+1} \\
F_{2n+2} &= F_{n-1}^2 + F_n^2 + F_n(F_{n-1} + F_{n+1}) \\
F_{2n+2} &= F_{n-1}^2 + F_n(F_n + F_{n-1} + F_{n+1}) \\
F_{2n+2} &= F_{n-1}^2 + F_n(2F_{n+1}) \\
F_{2n+2} &= (F_{n+1} - F_n)^2 - 2F_nF_{n+1} \\
F_{2n+2} &= F_n^2 + F_{n+1}^2
\end{aligned}
$$

$$
\begin{aligned}
F_{2n+3} &= F_{2n+1} + F_{2n+2} \\
F_{2n+3} &= F_n(F_{n-1} + F_{n+1}) + F_n^2 + F_{n+1}^2 \\
F_{2n+3} &= F_n(F_{n-1} + F_{n+1} + F_n) + F_{n+1}^2 \\
F_{2n+3} &= F_n(2F_{n+1}) + F_{n+1}^2 \\
F_{2n+3} &= F_{n+1}(2F_n + F_{n+1}) \\
F_{2n+3} &= F_{n+1}(F_n + F_{n+2})
\end{aligned}
$$

This concludes the proof by induction.

7. The inefficient algorithm computes F_n recursively as $F_{n-1} + F_{n-2}$, then each of these recursively without storing intermediate results. Let's call the number of additions required to compute F_n as A_n. Clearly, $A_0 = A_1 = 0$, since the algorithm uses the base case values of $F_0 = F_1 = 1$ and does not need to perform

any additions in this case. Further, we can see that $A_n = A_{n-1} + A_{n-2} + 1$, because in order to compute F_n naively, the algorithm would have to compute F_{n-1} and F_{n-2} independently, which would respectively take A_{n-1} and A_{n-2} additions, and then finally perform an extra addition to sum these two values. It is easy to see via induction that $A_n = F_n - 1$. This concludes the proof.

9b. Let's refer to the number of ways in which the positive integer n can be written as a summation of positive integers (wherein ordering of summands is relevant) as $S(n)$. We can easily see that $S(1) = 1$. For any other n, the last summand in any such summation is one of $\{1, 2, \cdots n-1\}$. In case the last summand is k, the rest of the sum $n-k$ can be written (by definition) in $S(n-k)$ ways. Therefore, we can see that $S(n) = \sum_{k=1}^{n-1} S(n-k)$. Since the first term is $S(n-1)$ and the sum of the remaining terms is also $S(n-1)$ (by recursion), we get $S(n) = S(n-1) + S(n-1) = 2S(n-1)$. It is easy to see via induction that $S(n) = 2^{n-1}$. This concludes the proof.

10. If n is odd, wee see that

$$
\begin{aligned}
f(n+2) - f(n+1) &= 2f(n+1) + 1 - (2f(n)) \\
f(n+2) - f(n+1) &= 2(f(n+1) - f(n)) + 1 \\
f(n+2) - f(n+1) &= 2(2f(n) - f(n)) + 1 \\
f(n+2) - f(n+1) &= 2f(n) + 1
\end{aligned}
$$

If n is even, we see that

$$
\begin{aligned}
f(n+2) - f(n+1) &= 2f(n+1) - (2f(n) + 1) \\
f(n+2) - f(n+1) &= 2(f(n+1) - f(n)) - 1 \\
f(n+2) - f(n+1) &= 2(2f(n) + 1 - f(n)) - 1 \\
f(n+2) - f(n+1) &= 2f(n) + 1
\end{aligned}
$$

This concludes the proof for the first part. For the second part, where we want to find an expression for $f(n)$ as a function of n, we start by noting that $f(1) = 1$ (because this is given) and $f(2) = 2$ (by a simple application of the provided recurrence formula). Let's now consider $n = 2k$, an even integer, and observe what the recurrence formula teaches us.

$$\begin{aligned}
f(2k+1) &= 2f(2k)+1 \\
f(2k+1) &= 2(2f(2k-1))+1 \\
f(2k+1) &= 4f(2k-1)+1 \\
\therefore f(2k+1) &= (4^{k+1}-1)/3 \\
\therefore f(n) &= (2^{n+1}-1)/3
\end{aligned}$$

and

$$\begin{aligned}
f(2k) &= 2f(2k-1) \\
f(2k) &= 2(2f(2k-2)+1) \\
f(2k) &= 4f(2k-2)+2 \\
\therefore f(2k) &= 2(4^{k}-1)/3 \\
\therefore f(n) &= (2^{n+1}-2)/3
\end{aligned}$$

The last line of the above two derivations is based on the sum of the geometric series $\sum_{i=0}^{r} 4^r = (4^{r+1}-1)/3$. For the "even" series, this works out just cleanly because $f(2) = 2$.

11a. Any valid sequence must either end with n or not. If it does end with n, the previous term (and all other terms before it) in the sequence must be in the range $\{1 \cdots \lfloor n/2 \rfloor\}$. If it does not end with n, then all terms in the sequence must be in the range $\{1 \cdots n-1\}$. Therefore, $s(n) = s(n-1) + s(\lfloor n/2 \rfloor)$. $s(0) = 1$ because the only possible sequence is the empty sequence. Following table shows some initial values.

n	s(n)
0	1
1	2
2	4
3	6
4	10
5	14
6	20
7	26
8	36
9	46
10	60

Now, let's consider the generating function $G(t) = \sum_{k=0}^{\infty} s(k)t^k$. Using the above recursion, we can express it as follows.

$$G(t) = s(0) + \sum_{k=1}^{\infty} s(k-1)t^k + \sum_{k=1}^{\infty} s(\lfloor k/2 \rfloor)t^k$$

For the first infinite sum on the RHS, we can replace $k-1$ by another variable l and get $\sum_{l=0}^{\infty} s(l)t^{l+1} = tG(t)$.

For the second infinite sum on the RHS, we need to do some more manipulation to simplify it.

$$\sum_{k=1}^{\infty} s(\lfloor k/2 \rfloor)t^k = s(0)t + s(1)t^2 + s(1)t^3 + s(2)t^4 + s(2)t^5 ...$$
$$\sum_{k=1}^{\infty} s(\lfloor k/2 \rfloor)t^k = (s(0)t + s(1)t^3 + s(2)t^5 + ...) + (s(1)t^2 + s(2)t^4 + ...)$$
$$\sum_{k=1}^{\infty} s(\lfloor k/2 \rfloor)t^k = t(s(0) + s(1)t^2 + s(2)t^4 + ...) + (-s(0)) + (s(0) + s(1)t^2 + s(2)t^4 + ...)$$
$$\sum_{k=1}^{\infty} s(\lfloor k/2 \rfloor)t^k = tG(t^2) + (-s(0)) + G(t^2)$$
$$\sum_{k=1}^{\infty} s(\lfloor k/2 \rfloor)t^k = (1+t)G(t^2) - s(0)$$

Combining everything, we get

$$G(t) = s(0) + tG(t) + (1+t)G(t^2) - s(0)$$
$$\therefore (1-t)G(t) = (1+t)G(t^2)$$

This concludes the proof.