Write the recurrence relation for the function definition to the right.

\[
R(n) = \begin{cases} 
\text{___________________________} & \text{if } n \text{ ______} \\
\text{___________________________} & \text{if } n \text{ ______} 
\end{cases}
\]

Calculate the first 6 terms of \(R(n)\). Then to the right calculate the sequence of differences between these terms. (You do not necessarily have to use all the slots for the sequences of differences - use what you need.)

\[\begin{array}{c|c|c|c|c|c|c|c|c|c|c}
\text{n} & 0 & 1 & 2 & 3 & 4 & 5 \\
\hline
\text{Sequences of differences} & & & & & & \\
\end{array}\]

Based on the sequence of differences (above) what is a good guess for the closed-form solution to the recurrence relation above?

\[f(n) = \underline{\text{___________}}\]

\[\begin{array}{c}
\text{A) } n + 2n + 1 \\
\text{B) } n^3 + n + 1 \\
\text{C) } n^4 + n^2 + 1 \\
\text{D) } n^2 + n + 1 \\
\text{E) } 2n + n + 1 \\
\text{F) } 2^n + n + 1 \\
\end{array}\]

Why?:

Verify this with a proof by induction. Prove \(R(n) = f(n)\) for all \(n \text{ ______} \).

Proof (Induction on \(n\)):

\[\begin{array}{c}
\text{: If } n = \underline{\text{____}}, \text{ the recurrence relation says } R(\underline{\text{____}}) = \underline{\text{____}}, \\
\text{and the closed-form solution says } f(\underline{\text{____}}) = \underline{\text{____}} = \underline{\text{____}}, \text{ so } R(\underline{\text{____}}) = f(\underline{\text{____}}). \\
\end{array}\]

\[\begin{array}{c}
\text{: Suppose as inductive hypothesis that } R(k-1) = \underline{\text{____}} \text{ for some } k > \underline{\text{____}}. \\
\end{array}\]

\[\begin{array}{c}
\text{: Using the recurrence relation, } R(k) = \underline{\text{____}}, \text{ by } 2^{\text{nd}} \text{ part of RR} \\
\text{= } \underline{\text{____}}, \text{ by IHOP} \\
\text{= } \underline{\text{____}} \\
\text{= } \underline{\text{____}} \\
\end{array}\]

So, by induction, \(R(n) = f(n) = \underline{\text{____}} \text{ for all } n \underline{\text{____}} \) (as desired).
Which of the general types of decompositions of recursive algorithms discussed in class would be good to use for each of the following?

Check if a string is a palindrome

Calculate the factorial of \( n \) as a recursive computer program taking a single formal parameter

Reverse chars in an array in place

Build a fractal like Koch snowflake

Binary search and merge sort

Reverse a string by moving last element to front, recurse on rest

\((sa)^R \rightarrow a(s)^R\)

Every year, Alice gets a raise of $3,000 plus 5% of her previous year's salary. Her starting salary is $50,000. Give a recurrence relation for \( S(n) \), Alice's salary after \( n \) years, for \( n \geq 0 \).

\[
S(n) = \begin{cases} 
\text{___________________________} & \text{if } n \text{ ______} \\
\text{___________________________} & \text{if } n \text{ ______} 
\end{cases}
\]

Match the person to what the person is famous for.

_____ Merge sort algorithm

_____ First Turing Award winner

_____ Credited as being the theoretical father of computer virology for his design of a self-reproducing program

_____ Credited as being the father of Computer Science

_____ A theoretical device representing a computing machine to understand limits of computation named after