Concepts:

- Define linear homogeneous recurrence relations of degree k with constant coefficients.
- Distinguish between linear versus nonlinear, and homogeneous versus nonhomogeneous recurrence relations.
- Compute a closed form solution of a linear homogeneous recurrence relation with constant coefficients.
- Prove statements about recursively defined sequences using induction.
- Set up recurrence relations in application problems.

Problems:

1. Complete the following statements.

   (a) Let B and C be real numbers. The characteristic equation of the linear recurrence relation of degree two of the form $a_n = Ba_{n-1} + Ca_{n-2}$ is ...

   (b) Let B and C be real numbers. If the characteristic equation of the linear recurrence relation of degree two of the form $a_n = Ba_{n-1} + Ca_{n-2}$ has two distinct real zeros r_1 and r_2, then the general solution is ...

   (c) Let B and C be real numbers. If the characteristic equation of the linear recurrence relation of degree two of the form $a_n = Ba_{n-1} + Ca_{n-2}$ has a repeated root r_0, then the general solution is ...

2. Characterize the following recurrence relations as linear homogeneous, linear nonhomogeneous, or nonlinear; with constant or nonconstant coefficients, and find their degrees.

   (a) $a_n = 6a_{n-1} + 3a_{n-2} - 4a_{n-4}$.

   (b) $a_{n+1} = n^2 a_{n-1} + na_{n-2} - 2^n a_{n-3}$.

   (c) $a_{n+2} = n^2 a_{n-1} + na_{n-2} - 2^n$.

   (d) $a_n = a_{n-1}a_{n-2} - 3a_{n-4}$.

3. True or False? The degree of the linear recurrence relation and the degree of the corresponding characteristic polynomial are equal.

4. What is the general solution of the linear recurrence relation with the given characteristic equation:

   (a) $r^2 - 2 = 0$.

   (b) $r^2 + 4 = 0$.

   (c) $(r + 9)^2 = 0$.

   (d) $(r - 2)^2(r + 3) = 0$.

(e) $(r+1)(r-2)(r+2) = 0$.

5. Find a closed-form solution of the following recurrence relations. Simplify your answer.

   (a) $a_n = 8a_{n-1} - 16a_{n-2}$ for $n \geq 2$ with initial conditions $a_0 = 4$ and $a_1 = 6$.

   (b) $a_n = 5a_{n-1} - 6a_{n-2}$ for $n \geq 2$ with initial conditions $a_0 = 2$ and $a_1 = 8$.

   (c) $a_n = -4a_{n-2}$ for $n \geq 2$ with initial conditions $a_0 = 0$ and $a_1 = 2$.

      i. Write the closed-form solution using complex numbers.

      ii. Reduce the solution from i. to a piece-wise formula that only contains real numbers.

      iii. Use both formulas from i. and ii. to find a_6 and a_7, and give your answer as a real number.

6. Suppose A is the set of bit strings recursively defined by

$$\begin{aligned} 10 &\in A \\ b \in A &\rightarrow 111b \in A \\ b \in A &\rightarrow 0b \in A. \end{aligned}$$

   Let a_n be the number of bit strings in A of length n, for $n \geq 0$. Determine a_0, a_1, a_2 and a recurrence relation for $n \geq 3$. Make sure to justify your recurrence relation carefully. In particular, you must make it clear that you are not double-counting bit strings.

7. Suppose B is the set of bit strings recursively defined by

$$\begin{aligned} 10 &\in B \\ b \in B &\rightarrow 111b \in B \\ b \in B &\rightarrow 1b \in B. \end{aligned}$$

   Let b_n be the number of bit strings in B of length n, for $n \geq 0$. Determine b_0, b_1 and b_2.

   This problem superficially looks very similar to the previous problem, only the 2nd recursion rule is slightly different. Can we use the same reasoning that we used in the previous problem to find a recurrence relation for b_n for $n \geq 3$?

   Is it true that $b_n = b_{n-1} + b_{n-3}$ for $n \geq 3$? Explain your answer.

8. A ternary string is a finite sequence of characters of 0's, 1's and 2's. Let a_n be the number of ternary strings of length n which do not contain the strings 12 or 11.

   (a) Evaluate a_1 and a_2.

   (b) Give a recurrence relation for a_n in terms of previous terms for $n \geq 3$. Explain how you set up the recurrence relation.

9. Let the sequence $\{a_n\}$ be defined recursively by $a_1 = 3, a_n = 2a_{n-1} + 2^{n-1}$ for all positive integers n. Use induction to prove that $a_n = 2^n + n2^{n-1}$ for all positive integers n.

Application Challenge Problems:

10. Let a_n be the number of ways a person can climb n stairs, if this person can only take only 1 or 2 stairs at a time.

    (a) Evaluate a_1 and a_2 and give a brief explanation.

    (b) Give a recurrence relation for a_n in terms of previous terms for $n \geq 2$.

    (c) Find the general solution for this linear recurrence relation.

    (d) Find a closed-form solution for a_n.

11. Let $T(n)$ be the number of arithmetic operations (additions, subtractions, multiplications) used in a "naive recursive calculation" of a_n by the recursive formula $a_n = 5a_{n-1} - 6a_{n-2}$, $n \geq 2$, with initial conditions $a_0 = 2$ and $a_1 = 8$. Use induction to show that $T(n) \geq 1.5^{n-1}$ for $n \geq 2$. (We assume that the recursive calculation does not store the already calculated values.)

12. The closed-form solution for $a_n = 5a_{n-1} - 6a_{n-2}$, $n \geq 2$, with initial conditions $a_0 = 2$ and $a_1 = 8$ is $a_n = (-2) \cdot 2^n + 4 \cdot 3^n$. (See Problem 5.(b)).

    (a) Let us assume that you work for a company, and your task is find the values of a_{10} and a_{50}.
    What formula would you use in your algorithm, the recursive formula $a_n = 5a_{n-1} - 6a_{n-2}$ or the closed-form formula $a_n = (-2) \cdot 2^n + 4 \cdot 3^n$? Explain your answer.

    (b) If one operation requires only a nanosecond, at least how many years will the recursive algorithm take to complete for $n = 100$?

    (c) Write two short programs which calculate a_{10} and a_{50}. Use the recursive formula for the first, and the closed-form formula for the second program. What did you notice between the two approaches?

Solutions:

1. Complete the following statements.

   (a) Let B and C be real numbers. The characteristic equation of the linear recurrence relation of degree two of the form $a_n = Ba_{n-1} + Ca_{n-2}$ is $r^2 - Br - C = 0$.

   (b) Let B and C be real numbers. If the characteristic equation of the linear recurrence relation of degree two of the form $a_n = Ba_{n-1} + Ca_{n-2}$ has two distinct real zeros r_1 and r_2, then the general solution is $a_n = \alpha_1 r_1^n + \alpha_2 r_2^n$ where α_1 and α_2 are arbitrary real numbers.

   (c) Let B and C be real numbers. If the characteristic equation of the linear recurrence relation of degree two of the form $a_n = Ba_{n-1} + Ca_{n-2}$ has a repeated root r_0, then the general solution is $a_n = \alpha_1 r_0^n + \alpha_2 n r_0^n$ where α_1 and α_2 are arbitrary real numbers.

2. Characterize the following recurrence relations as linear homogeneous, linear nonhomogeneous, or nonlinear; with constant or nonconstant coefficients, and find their degrees.

   (a) $a_n = 6a_{n-1} + 3a_{n-2} - 4a_{n-4}$.

   This is a linear homogeneous recurrence relation of degree 4 with constant coefficients. It is homogeneous because all terms are of the form $f(n)a_j$.

   (b) $a_{n+1} = n^2 a_{n-1} + na_{n-2} - 2^n a_{n-3}$.

   This is a linear homogeneous recurrence relation of degree 4 with nonconstant coefficients. It is homogeneous because all terms are of the form $f(n)a_j$.

   (c) $a_{n+2} = n^2 a_{n-1} + na_{n-2} - 2^n$.

   This is a linear nonhomogeneous recurrence relation of degree 4 with nonconstant coefficients. It is nonhomogeneous because of the 2^n term.

   (d) $a_n = a_{n-1}a_{n-2} - 3a_{n-4}$.

   This is a nonlinear recurrence relation of degree 4 with constant coefficients. It is nonlinear because of the $a_{n-1}a_{n-2}$ term.

3. True or False? The degree of the linear recurrence relation and the degree of the corresponding characteristic polynomial are equal.

   True.

4. What is the general solution of the linear recurrence relation with characteristic equation:

   (a) $r^2 - 2 = 0$.

   $a_n = \alpha_1(\sqrt{2})^n + \alpha_2(-\sqrt{2})^n$, where α_1 and α_2 are arbitrary real numbers.

(b) $r^2 + 4 = 0$.

$a_n = \alpha_1(2i)^n + \alpha_2(-2i)^n$, where α_1 and α_2 are complex conjugates.

(c) $(r + 9)^2 = 0$.

$a_n = \alpha_1(-9)^n + \alpha_2 n(-9)^n$, where α_1 and α_2 are arbitrary real numbers.

(d) $(r - 2)^2(r + 3) = 0$.

$a_n = \alpha_1 2^n + \alpha_2 n 2^n + \alpha_3(-3)^n$, where α_1, α_2 and α_3 are arbitrary real numbers.

(e) $(r + 1)(r - 2)(r + 2) = 0$.

$a_n = \alpha_1(-1)^n + \alpha_2 2^n + \alpha_3(-2)^n$, where α_1, α_2 and α_3 are arbitrary real numbers.

5. Find a closed-form solution of the following recurrence relations. Simplify your answer.

(a) $a_n = 8a_{n-1} - 16a_{n-2}$ for $n \geq 2$ with initial conditions $a_0 = 4$ and $a_1 = 6$.

The characteristic equation is $r^2 - 8r + 16 = 0$. Thus, $r_1 = r_2 = 4$ and the general solution is $a_n = p \cdot 4^n + q \cdot n \cdot 4^n$ where p, q are arbitrary real numbers.

Using the given initial conditions we obtain:
$a_0 = 4 = p + 0$ and $a_1 = 6 = 4p + 4q$.
Thus, $p = 4$ and $q = -2.5$, and the closed form solution is $a_n = 4 \cdot 4^n - 2.5 \cdot n \cdot 4^n = 4^{n+1} - 2.5 \cdot n \cdot 4^n$.

(b) $a_n = 5a_{n-1} - 6a_{n-2}$ for $n \geq 2$ with initial conditions $a_0 = 2$ and $a_1 = 8$.

The characteristic equation is $r^2 - 5r + 6 = 0$. Thus, $r_1 = 2$ and $r_2 = 3$ and the general solution is $a_n = p \cdot 2^n + q \cdot 3^n$ where p, q are arbitrary real numbers.

Using the given initial conditions we obtain:
$a_0 = 2 = p + q$ and $a_1 = 8 = 2p + 3q$.
Thus, $p = -2$ and $q = 4$ and the closed form solution is $a_n = (-2) \cdot 2^n + 4 \cdot 3^n = -2^{n+1} + 4 \cdot 3^n$.

(c) $a_n = -4a_{n-2}$ for $n \geq 2$ with initial conditions $a_0 = 0$ and $a_1 = 2$.

i. Write the closed-form solution using complex numbers.

The characteristic equation is $r^2 + 4 = 0$. Thus, $r_1 = -2i$ and $r_2 = 2i$ and the general solution is $a_n = (a + bi) \cdot (-2i)^n + (a + bi) \cdot (2i)^n$ with arbitrary real numbers a, b.

Using the given initial conditions we obtain:
$a_0 = 0 = (a + bi) + (a - bi) = 2a$ and $a_1 = 2 = (a + bi)(-2i) + (a - bi)(2i)$.
From the first equation we obtain $a = 0$, and from the second equation we derive that $b = \frac{1}{2}$.
Hence the closed form solution is $a_n = (\frac{i}{2}) \cdot (-2i)^n + (-\frac{i}{2}) \cdot (2i)^n$.

ii. Reduce the solution from i. to a piece-wise formula that only contains real numbers.

Applying the laws of exponentiation, we obtain

$a_n = (\frac{i}{2}) \cdot (-2i)^n + (-\frac{i}{2}) \cdot (2i)^n = (-1)^n i^{n+1} 2^{n-1} - i^{n+1} 2^{n-1} = i^{n+1} 2^{n-1}((-1)^n - 1)$.

If n is even, $(-1)^n = 1$, and then $a_n = i^{n+1} 2^{n-1}((-1)^n - 1) = 0$.

If n is odd, $(-1)^n = -1$, and then $a_n = i^{n+1} 2^{n-1}((-1)^n - 1) = -i^{n+1} 2^n = -(i^2)^{\frac{n+1}{2}} 2^n =$

$-(-1)^{\frac{n+1}{2}} 2^n = (-1)^{\frac{n+3}{2}} 2^n = (-1)^{\frac{n-1}{2}} 2^n$. Note that, when n is odd, $(-1)^{\frac{n+3}{2}} = (-1)^{\frac{n-1}{2}}$. Thus,

$$a_n = \begin{cases} 0 \text{ if } n \text{ is even} \\ (-1)^{\frac{n-1}{2}} \cdot 2^n \text{ if } n \text{ is odd.} \end{cases}$$

iii. Use both formulas from i. and ii. to find a_6 and a_7, and give your answer as a real number.

First formula: $a_6 = (\frac{i}{2}) \cdot (-2i)^6 + (-\frac{i}{2}) \cdot (2i)^6 = (-32i) + (32i) = 0$.
Second formula: Since 6 is even, $a_6 = 0$.

First formula: $a_7 = (\frac{i}{2}) \cdot (-2i)^7 + (-\frac{i}{2}) \cdot (2i)^7 = (-64) + (-64) = -128$.
Second formula: $a_7 = (-1)^{\frac{7-1}{2}} \cdot 2^7 = -128$.

6. Suppose A is the set of bit strings recursively defined by

$$\begin{aligned} &10 \in A \\ &b \in A \to 111b \in A \\ &b \in A \to 0b \in A. \end{aligned}$$

Let a_n be the number of bit strings in A of length n, for $n \geq 0$. Determine a_0, a_1, a_2 and a recurrence relation for $n \geq 3$. Make sure to justify your recurrence relation carefully. In particular, you must make it clear that you are not double-counting bit strings.

$a_0 = 0$, $a_1 = 0$ and $a_2 = 1$, since 10 is only one element of length 2 and there are no elements of length 1 and 0 in the set.

For $n \geq 3$, any bit string c of length n in A is generated recursively by one of the following mutually exclusive alternatives:

Case 1: $c = 111b$, where b is a bit string of length $n-3$ in A. There are a_{n-3} such bit strings.

Case 2: $c = 0b$, where b is a bit string of length $n-1$ in A. There are a_{n-1} such bit strings.

Case 2 only generates bit strings starting with 0, which can not be generated in Case 1. Furthermore any bit string generated in Case 1 can not be generated in Case 2. Thus, Case 1 and Case 2 are mutually exclusive, so we don't generate bit strings multiple times. There is exactly one path to get to each element of A in the recursion tree from the initial element.

Hence, $a_n = a_{n-1} + a_{n-3}$, $n \geq 3$, with initial conditions $a_0 = 0, a_1 = 0$ and $a_2 = 1$. Note that this is a linear homogeneous recurrence relation of degree 3.

7. Suppose B is the set of bit strings recursively defined by:

$$\begin{aligned} &10 \in B \\ &b \in B \to 111b \in B \\ &b \in B \to 1b \in B. \end{aligned}$$

Let b_n be the number of bit strings in B of length n, for $n \geq 0$. Determine b_0, b_1 and b_2.

This problem superficially looks very similar to the previous problem, only the 2nd recursion rule is slightly different. Can we use the same reasoning that we used in the previous problem to find a recurrence relation for b_n for $n \geq 3$?

Is it true that $b_n = b_{n-1} + b_{n-3}$ for $n \geq 3$? Explain your answer.

$b_0 = 0$, $b_1 = 0$ and $b_2 = 1$, since 10 is only one element of length 2 and there are no elements of length 1 and 0 in the set.

We can not use the same idea. In the previous problem, a string in the set A could be generated only one way. In this problem the same strings in B can be generated multiple times. That is, there are multiple paths to get to some elements of B in the recursion tree.

For example, the string 11110 can be generated by applying the first rule on 10, or by applying the 2nd rule three times in cycles. Therefore the recursive rules are not mutually exclusive in this problems.

The formula $b_n = b_{n-1} + b_{n-3}$ already fails for $n = 5$. 11110 is the only string of length 5 in B, so $b_5 = 1$. According to the formula $b_5 = b_4 + b_2$, b_5 should be 2, since $b_2 = 1$ and $b_4 = 1$. Note that 1110 is the only string in B with length 4.

The recursive formula $b_n = b_{n-1} + b_{n-3}$ counts many elements multiple times, so $b_n = b_{n-1} + b_{n-3}$ is not true for all $n \geq 3$.

8. A ternary string is a finite sequence of characters of 0's, 1's and 2's. Let a_n be the number of ternary strings of length n which do not contain the strings 12 or 11.

   (a) Evaluate a_1 and a_2.

   $a_1 = 3$, since there are three ternary strings of length one, 0, 1 and 2, which do not contain the strings 12 or 11. The following ternary strings of length two 00, 01, 10, 02, 20, 22 and 21 don't contain the strings 12 or 11, thus $a_2 = 7$.

   (b) Give a recurrence relation for a_n in terms of previous terms for $n \geq 3$. Explain how you set up the recurrence relation.

   If $n \geq 3$, then we can generate any such bit string c recursively by considering the following three alternatives:

   Case 1: if $c = 0b$, where b is a ternary string of length $n-1$ not containing 12 or 11. There are a_{n-1} ways to choose b.
   Case 2: if $c = 2b$, where b is a ternary string of length $n-1$ not containing 12 or 11. There are a_{n-1} ways to choose b.
   Case 3: if $c = 10b$, where b is a ternary string of length $n-2$ not containing 12 or 11. There are a_{n-2} ways to choose b.

   Thus, $a_n = 2a_{n-1} + a_{n-2}$ for $n \geq 3$. We counted all the possibilities, since a ternary string not containing 12 or 11 either starts with a 0, or a 2, or 10, there are no other options. We did not double count any possibilities, since these cases are mutually exclusive.

   Note that this is a second order (degree 2) linear homogeneous relation with constant coefficients.

9. Let the sequence $\{a_n\}$ be defined recursively by $a_1 = 3, a_n = 2a_{n-1} + 2^{n-1}$ for all positive integers n. Use induction to prove that $a_n = 2^n + n2^{n-1}$ for all positive integers n.

Proof:

Basis Step: The $n = 1$ case is true, since $a_1 = 3$ by definition and $2^n + n2^{n-1} = 3$ for $n = 1$.

Inductive Step: Assume that $a_n = 2^n + n2^{n-1}$ has been proved for an arbitrary positive integer n.

We wish to prove that $a_{n+1} = 2^{n+1} + (n+1) \cdot 2^n$.

Using the inductive hypothesis and the recursive definition,
$a_{n+1} = 2a_n + 2^n = 2(2^n + n2^{n-1}) + 2^n = 2^{n+1} + n2^n + 2^n = 2^{n+1} + (n+1) \cdot 2^n$ which shows that a_{n+1} has the required form.

The proof is completed by induction.

Application Challenge Problems:

10. Let a_n be the number of ways a person can climb n stairs, if this person can only take 1 or 2 stairs at a time.

(a) Evaluate a_1 and a_2 and give a brief explanation.

$a_1 = 1$. A person can climb 1 stair only in 1 way.
$a_2 = 2$. A person can climb 2 stairs in two ways, stepping 1 stair twice or stepping 2 stairs at once.

(b) Give a recurrence relation for a_n in terms of previous terms for $n \geq 2$. Explain how you get your recurrence relation.

If $n \geq 3$, then a person can climb n stairs the following two ways:

Case 1: start with a step of 1 and climb the rest of the $n-1$ stairs, which can be climbed a_{n-1} ways.

Case 2: start with a step of 2 stairs and climb the rest of the $n-2$ stairs, which can be climbed a_{n-2} ways.

Thus, $a_n = a_{n-1} + a_{n-2}$ for $n \geq 3$. A person can start the climbing with either stepping 1 stair or stepping 2 stairs, therefore we counted all the possibilities, and we did not double count any possibilities, since the cases are exclusive.

(c) Find the general solution for this linear recurrence relation.

The characteristic equation is $r^2 - r - 1 = 0$. Thus, $r_1 = \frac{1-\sqrt{5}}{2}$ and $r_2 = \frac{1+\sqrt{5}}{2}$.
The general solution is $a_n = p \cdot (\frac{1-\sqrt{5}}{2})^n + q \cdot (\frac{1-\sqrt{5}}{2})^n$, where p, q are arbitrary real numbers.

(d) Find a closed-form solution for a_n.

Let us define $a_0 = 1$. Note that $a_0 + a_1 = a_2$ is satisfied.
Using the given initial conditions we obtain:
$a_0 = 1 = p + q$
$a_1 = 1 = p(\frac{1-\sqrt{5}}{2}) + q(\frac{1+\sqrt{5}}{2})$
Thus, $p = \frac{\sqrt{5}-1}{2\sqrt{5}}$ and $q = \frac{\sqrt{5}+1}{2\sqrt{5}}$ and the closed form solution is
$a_n = (\frac{\sqrt{5}-1}{2\sqrt{5}}) \cdot (\frac{1-\sqrt{5}}{2})^n + (\frac{\sqrt{5}+1}{2\sqrt{5}}) \cdot (\frac{1+\sqrt{5}}{2})^n$

Note the a_n is a shifted Fibonacci sequence. In fact $a_n = f_{n+1}$ for $n \geq 0$.

$a_0 = 1, a_1 = 1, a_2 = 2, a_3 = 3, a_4 = 5, a_5 = 8\ldots$ and
$f_0 = 0, f_1 = 1, f_2 = 1, f_3 = 2, f_4 = 3, f_5 = 5, f_6 = 8\ldots$

11. Let $T(n)$ be the number of arithmetic operations (additions, subtractions, multiplications) used in a "naive recursive calculation" of a_n by the recursive formula $a_n = 5a_{n-1} - 6a_{n-2}$, $n \geq 2$, with initial conditions $a_0 = 2$ and $a_1 = 8$. Use induction to show that $T(n) > 1.5^{n-1}$ for $n \geq 2$. (We assume that the recursive calculation does not store the already calculated values.)

    Proof: First we will set up a recurrence relation for $T(n)$. It takes $T(n-1)$ operations to calculate a_{n-1} and $T(n-2)$ operations for a_{n-2}. In addition 2 extra multiplications and a subtraction are required to find the value of a_n. Thus, $T(n) = T(n-1) + T(n-2) + 3$, $n \geq 2$, with initial conditions $T(0) = 0$ and $T(1) = 0$.

    (Note that this is a second order linear nonhomogeneous relation with constant coefficients and the exact solution can be found in a systematic way as discussed in the book.)

    Basis Step: In order to verify the base case, we need to show that $T(n) > 1.5^{n-1}$ for $n = 2$ and $n = 3$.

    For $n = 2$, $T(2) = T(1) + T(0) + 3 = 3$ and $3 > 1.5^{2-1}$,
    and for $n = 3$, $T(3) = T(2) + T(1) + 3 = 6$ and $6 > 1.5^{3-1} = 2.25$.

    Inductive Step: Assume $T(n) > 1.5^{n-1}$ and $T(n-1) > 1.5^{n-2}$ have been proved for an arbitrary integer n, $n \geq 3$.

    Now we wish to show that $T(n+1) \geq 1.5^n$.

    According to the recursive formula for $T(n+1)$ and the inductive hypothesis,

    $T(n+1) = T(n) + T(n-1) + 3 > 1.5^{n-1} + 1.5^{n-2} + 3 > 1.5^{n-1} + 1.5^{n-2} = 1.5^{n-1}(1 + 1.5^{-1}) =$

    $1.5^{n-1}(1 + \frac{2}{3}) = 1.5^{n-1} \cdot \frac{5}{3} > 1.5^{n-1} \cdot \frac{3}{2} = 1.5^n$.

    Thus, by the Principle of Mathematical Induction we have proved that $T(n) > 1.5^{n-1}$ for $n \geq 2$, i.e., calculating a_n by the given recursive formula requires more than 1.5^n arithmetic operations for $n \geq 2$.

12. The closed-form solution for $a_n = 5a_{n-1} - 6a_{n-2}$, $n \geq 2$, with initial conditions $a_0 = 2$ and $a_1 = 8$ is $a_n = (-2) \cdot 2^n + 4 \cdot 3^n$. (See Problem 5.(b)).

    (a) Let us assume that you work for a company, and your task is find the values of a_{10} and a_{50}.
    What formula would you use in your algorithm, the recursive formula $a_n = 5a_{n-1} - 6a_{n-2}$ or the closed-form formula $a_n = (-2) \cdot 2^n + 4 \cdot 3^n$? Explain your answer.

    The closed-form formula will calculate a_n much more efficiently than the recursive formula as n increases. According to the previous problem, $T(n) > 1.5^{n-1}$. This means the number of arithmetic operations, that are used in the calculation of a_n by the recursive formula, grows exponentially as n increases. The running time of this algorithm would exceed the estimated life time of the universe even for relatively small values of n.
    On the other hand the closed-form formula $a_n = (-2) \cdot 2^n + 4 \cdot 3^n$ uses roughly $2n + 1$ arithmetic operations, $2n$ multiplications and an addition, which has linear running time.

(b) If one operation requires only a nanosecond, at least how many years will the recursive algorithm take to complete for $n = 100$?

For $n = 100$, the recursive formula uses more than $1.5^{99} \approx 2.7 \cdot 10^{17}$ operations. (For comparison notice that the closed-form formula uses about $2 \cdot 100 + 1 = 201$ operations.)

Thus, with recursive formula, it would take more than $\frac{2.7 \cdot 10^{17} \cdot 10^{-9}}{60 \cdot 60 \cdot 24 \cdot 365.25} \approx 8.56$ years to calculate a_{100}.

(c) Write two short programs which calculate a_{10}, a_{50}. Use the recursive formula for the first, and the closed-form formula for the second program. What did you notice between the two approaches?

The following Python program calculates a_n recursively and outputs a_n for a hard-coded value of n (here $n = 10$) followed by the time, in seconds, the computation required.

```
import time

def a(n):
        if n == 0:
                return 2
        elif n == 1:
                return 8
        else:
                return 5*a(n-1)-6*a(n-2)

start_time = time.time()
print(a(10))
end_time  = time.time()
print(end_time - start_time)
```

The output of the program is 234148, $5.1975250244140625e - 05$,

i.e. $a_{10} = 234148$, and the computation was essentially instantaneous.

If we alter the third line of the program to print(a(50)), the output is 2871591948515610541395748, 4046.326828479767

i.e. $a_{50} = 2871591948515610541395748$, and the computation took over 4046 seconds, or over 67 minutes.

The following Python program computes a_n, for a hard-coded value of n (here $n = 10$) using the closed form of the sequence a_n:

```
import time

start_time = time.time()
n = 10
a = (-2)*2**n + 4*3**n
print(a)
end_time  = time.time()
print(end_time - start_time)
```

It produces the output 234148, $3.457069396972656e - 05$.

We see that the computed value for $a_{10} = 234148$ agrees with our recursive computation, and the calculation was once again virtually instantaneous, unsurprisingly.

If we change the third line of the program to $n = 50$, the dramatic improvement in run time becomes apparent. The program then produces the output 2871591948515610541395748, $3.0994415283203125e - 05$.

Again, we find that $a_{50} = 2871591948515610541395748$, but this time, the calculation was near-instantaneous, instead of taking over an hour. If we take the measured times at face value, this calculation was about

$1.3 \cdot 10^8$ or over 100 million times faster.

Note that these time values depend on the computer used, and even vary when the program is re-run on the same computer.