Algorithm Hw 3 Chih-Hsiang Wang 933271081

1. for i = 1 to 2n-1 by step 2
   for j = 1 to i by step 1
   {computation}

outer loop	inner loop	# of {computation}
i = 1	j = 1 to 1	1
i = 3	j = 1 to 3	3
i = 5	j = 1 to 5	5
	⋮	
i = 2n-1	j = 1 to 2n-1	2n - 1

so total times of {computation}

$\Rightarrow 1 + 3 + 5 + \cdots + 2n - 1 = (1 + 2n - 1) \times n / 2 = n^2$

2.

(a) $GCD(233, 144) = GCD(144, 233 \bmod 144) = GCD(144, 89) = GCD(89, 144 \bmod 89)$

$= GCD(89, 55) = GCD(55, 89 \bmod 55) = GCD(55, 34) = GCD(34, 55 \bmod 34)$

$= GCD(34, 21) = GCD(21, 34 \bmod 21) = GCD(21, 13) = GCD(13, 21 \bmod 13)$

$= GCD(13, 8) = GCD(8, 13 \bmod 8) = GCD(8, 5) = GCD(5, 8 \bmod 5) = GCD(5, 3)$

$= GCD(3, 5 \bmod 3) = GCD(3, 2) = GCD(2, 3 \bmod 2) = GCD(2, 1) = GCD(1, 2 \bmod 1)$

$= GCD(1, 0) \Rightarrow GCD(233, 144) = 1$

(b) $gcd(f_n, f_{n-1}) = gcd(f_{n-1} + f_{n-2}, f_{n-1}) = gcd(f_{n-1}, (f_{n-1} + f_{n-2}) \bmod f_{n-1}) = gcd(f_{n-1}, f_{n-2})$

$= gcd(f_{n-2} + f_{n-3}, f_{n-2}) = gcd(f_{n-2}, (f_{n-2} + f_{n-3}) \bmod f_{n-2}) = gcd(f_{n-2}, f_{n-3}) \ldots\ldots$

$\Rightarrow gcd(f_n, f_{n-1}) = gcd(2, 1) = 1$

(c) $GCD(f_n, f_{n-1}) = \begin{cases} 1 & , \text{for } n = 1 \\ GCD(f_n, f_{n-1}) & , \text{for } n > 1 \end{cases}$

number of recursive calls = n - 2, for > 2

(d) Base case : $GCD(f_1, f_0) = GCD(1, 0) = 1$, true

Hypothesis : assume true for $n = 1, 2, 3, \cdots k-1$, $k > 0$

$n = k-1 : GCD(f_{k-1}, f_{k-2}) = 1$, true

for $n = k : GCD(f_k, f_{k-1}) = GCD(f_{k-1}, f_{k-2}) = 1$

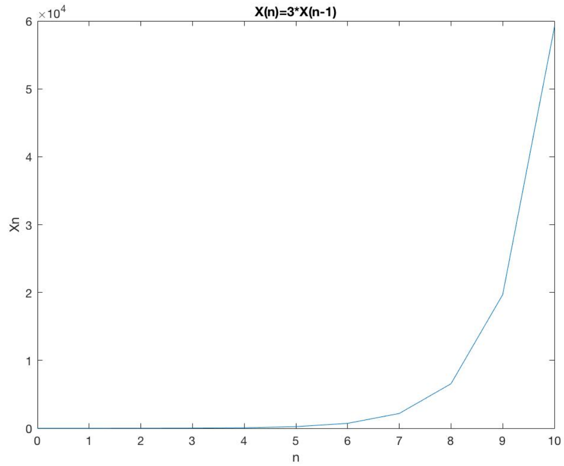
solved by induction steps

3、

(a) $x_n = 3x_{n-1}$, $x_0 = 1$, $\lambda - 3 = 0 \Rightarrow \lambda = 3 \Rightarrow x_n = c\cdot 3^n$, $x_0 = 1 = c\cdot 3^0 \Rightarrow c = 1$

$\Rightarrow x_n = 3^n$. $O(x_n) = O(3^n)$

Homogeneouse ($g(n) \equiv 0$), then $x_n = \Theta(\lambda^n) = \Theta(3^n)$, is the same as calculated

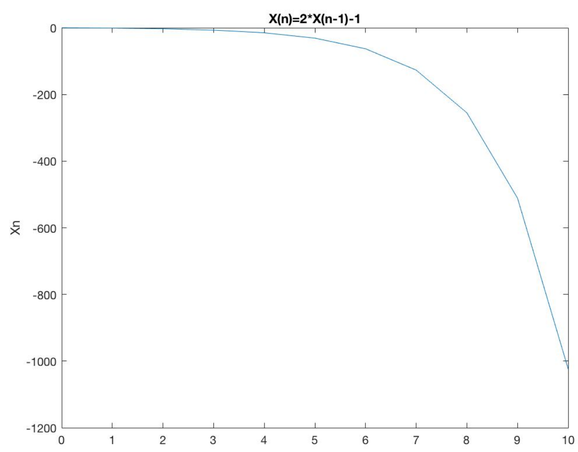


(b) $x_n = 2x_{n-1} - 1$, $x_0 = 0$, $\lambda_h - 2 = 0 \Rightarrow \lambda_h = 2 \Rightarrow x_n = a\cdot 2^n + \lambda_p = a\cdot 2^n + bn + c$

$x_0 = 0 = a\cdot 2^0 + b\cdot 0 + c = a + c$, $x_1 = -1 = a\cdot 2^1 + b\cdot 1 + c = 2a + b + c$, $x_2 = -3 = a\cdot 2^2 + b\cdot 2 + c = 4a + 2b + c$

$$\begin{cases} 0 = a + c \\ -1 = 2a + b + c \\ -3 = 4a + 2b + c \end{cases}$$

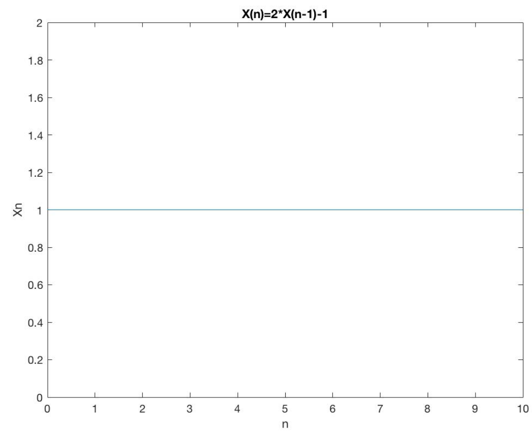
$a = -1$, $b = 0$, $c = 1$ $\Rightarrow x_n = -2^n + 1$, $O(x_n) = O(-2^n)$



(c) $X_n = 2X_{n-1} - 1$, $X_0 = 1$, $\lambda_h - 2 = 0 \Rightarrow \lambda_h = 2 \Rightarrow X_n = a \cdot 2^n + \lambda_p = a \cdot 2^n + bn + c$

$X_0 = 1 = a \cdot 2^0 + b \cdot 0 + c$, $X_1 = 1 = a \cdot 2^1 + b \cdot 1 + c$, $X_2 = 1 = a \cdot 2^2 + b \cdot 2 + c$

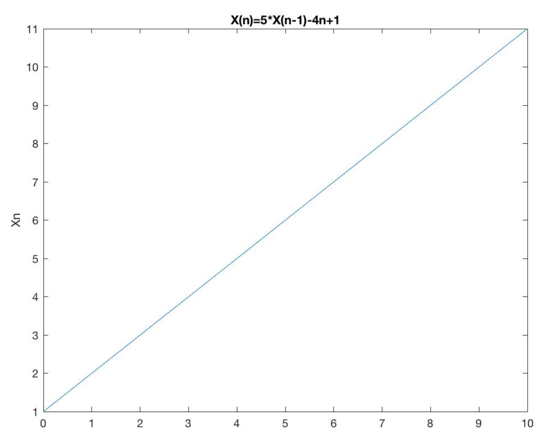
$\begin{cases} 1 = a + c \\ 1 = 2a + b + c \\ 1 = 4a + 2b + c \end{cases}$ $a = 0, b = 0, c = 0 \Rightarrow X_n = 1$. $O(X_n) = O(1)$



(d) $X_n = 5X_{n-1} - 4n + 1$, $X_0 = 1$, $\lambda_h - 5 = 0 \Rightarrow \lambda_h = 5 \Rightarrow X_n = a \cdot 5^n + \lambda_p = a \cdot 5^n + bn + c$

$X_0 = 1 = a \cdot 5^0 + b \cdot 0 + c$, $X_1 = 2 = a \cdot 5^1 + b \cdot 1 + c$, $X_2 = 3 = a \cdot 5^2 + b \cdot 2 + c$

$\begin{cases} 1 = a + c \\ 2 = 5a + b + c \\ 3 = 25a + 2b + c \end{cases}$ $a = 0, b = 1, c = 1 \Rightarrow X_n = n + 1$. $O(X_n) = O(n)$



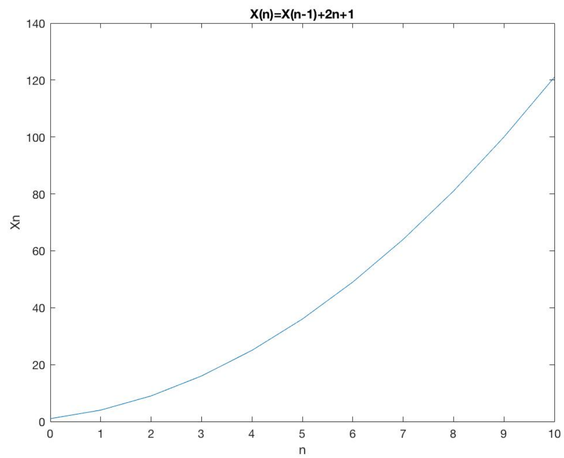
(e) $x_n = x_{n-1} + 2n + 1$, $x_0 = 1$ $\lambda_h - 1 = 0 \Rightarrow \lambda_h = 1 \Rightarrow x_n = an^2 + bn + c$

$x_0 = 1 = a \cdot 0^2 + b \cdot 0 + c$ $x_1 = 4 = a \cdot 1^2 + b \cdot 1 + c$ $x_2 = 9 = a \cdot 2^2 + b \cdot 2 + c$

$\begin{cases} 1 = c \\ 4 = a + b + 1 \\ 9 = 4a + 2b + 1 \end{cases}$ $a = 1$, $b = 2$, $c = 1 \Rightarrow x_n = n^2 + 2n + 1$. $O(x_n) = O(n^2)$

Non-Homogeneous $(g(n) > 0)$, $g(n) = 2n+1$, $\lambda = 1$.

$g(n) = \theta(n^1 \cdot 1^n)$, then $x_n = \theta(n^{1+1} \cdot 1^n) = \theta(n^2)$, is the same as calculated



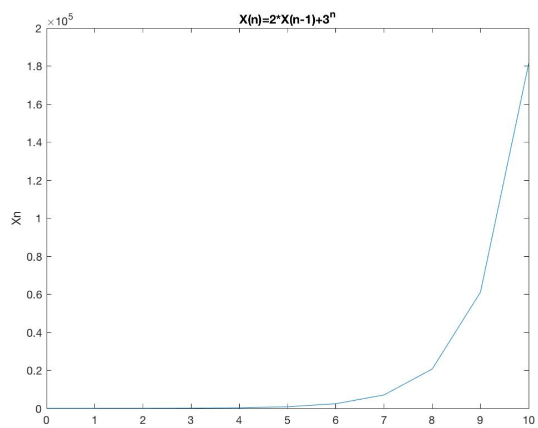
(f) $x_n = 2x_{n-1} + 3^n$, $x_0 = 7$ $\lambda_h - 2 = 0 \Rightarrow \lambda_h = 2 \Rightarrow x_n = a \cdot 2^n + \lambda_p = a \cdot 2^n + b \cdot 3^n$

$x_0 = 7 = a \cdot 2^0 + b \cdot 3^0$ $x_1 = 17 = a \cdot 2^1 + b \cdot 3^1$ $x_2 = 43 = a \cdot 2^2 + b \cdot 3^2$

$\begin{cases} 7 = a + b \\ 17 = 2a + 3b \\ 43 = 4a + 9b \end{cases}$ $a = 4$, $b = 3 \Rightarrow x_n = 4 \cdot 2^n + 3 \cdot 3^n$ $O(x_n) = O(3^n)$

Non-Homogeneous $(g(n) > 0)$, $g(n) = 3^n$, $\lambda = 2$

$g(n) = \theta(3^n)$, with $\gamma = 3 > \lambda = 2$, then $x_n = \theta(\gamma^n) = \theta(3^n)$ is the same as calculated



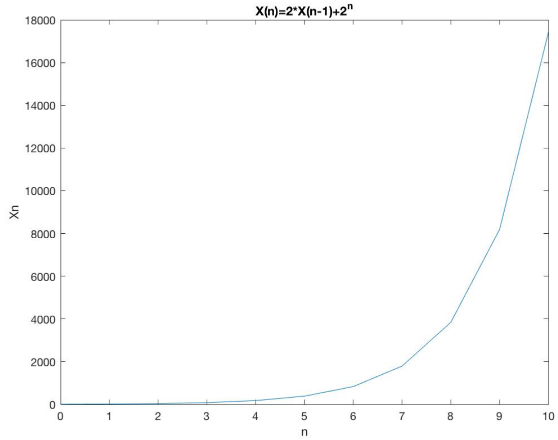
(9)' $X_n = 2X_{n-1} + 2^n$, $X_0 = 7$ $\quad \lambda_h - 2 = 0 \Rightarrow \lambda_h = 2 \Rightarrow X_n = a \cdot 2^n + \lambda_p = a \cdot 2^n + b \cdot n \cdot 2^n$

$X_0 = 7 = a \cdot 2^0 + b \cdot 0 \cdot 2^0$, $X_1 = 16 = a \cdot 2^1 + b \cdot 1 \cdot 2^1$, $X_2 = 36 = a \cdot 2^2 + b \cdot 2 \cdot 2^2$

$\begin{cases} 7 = a \\ 16 = 2a + 2b \\ 36 = 4a + 8b \end{cases}$ $\quad a = 7,\ b = 1 \Rightarrow X_n = 7 \cdot 2^n + n \cdot 2^n$. $O(X_n) = O(n \cdot 2^n)$

Non-Homogeneous ($g(n) > 0$), $g(n) = 2^n$, $\lambda = 2$

$g(n) = \Theta(n^0 \cdot 2^n)$, then $X_n = \Theta(n^{d+1} \lambda^n) = \Theta(n \cdot 2^n)$ is the same as calculated



4. assume $T(3) > 0$, $T(2) > 0$, $T(1) > 0$, need to show $T(n) = \Theta(\lambda_0^n)$ where $\lambda_0^3 = \lambda_0^2 + \lambda_0 + 1$

Because $T(n) = T(n-1) + T(n-2) + T(n-3)$ is homogeneous and non-negative

$\lambda_0^3 = \lambda_0^2 + \lambda_0 + 1 \Rightarrow$ we can assume $T(n) = c\lambda_0^n$

Base case: $T(1) = c \cdot \lambda_0^1 \Rightarrow \Theta(\lambda^1)$

Hypothesis: assume true for $n = 1, 2, 3, 4, \cdots k$, $k > 0$

$T(n+1) = T(n) + T(n-1) + T(n-2) = c\lambda_0^n + c\lambda_0^{n-1} + c\lambda_0^{n-2} = c\lambda_0^{n-2}(\lambda_0^2 + \lambda_0 + 1)$

we know $\lambda_0^3 = \lambda_0^2 + \lambda_0 + 1 \Rightarrow T(n+1) = c\lambda_0^{n-2} \cdot \lambda_0^3 = c\lambda_0^{n+1} \Rightarrow T(n) = \Theta(\lambda_0^{n+1})$

Proved by induction steps

5. min : $f(0) = 1$, $f(1) = 2$, $f(2) = 4$, $f(3) = 7 \ldots \Rightarrow f(h) = f(h-1) + f(h-2) + 1$

max : $g(0) = 1$, $g(1) = 3$, $g(2) = 7$, $g(3) = 15 \ldots \Rightarrow g(h) = 2^{h+1} - 1$

$f_h = f_{h-1} + f_{h-2} + 1$, $\lambda_h^2 - \lambda_h - 1 = 0$, $\lambda = \dfrac{1 \pm \sqrt{5}}{2} \Rightarrow f_n = a \cdot \left(\dfrac{1+\sqrt{5}}{2}\right)^h + b \cdot \left(\dfrac{1-\sqrt{5}}{2}\right)^h + cn + d$

$\lambda_p = cn + d \Rightarrow cn + d = c(n-1) + d + c(n-2) + d + 1$, $c = 0$, $d = -1$

$$\begin{cases} 1 = a + b - 1 \\ 2 = a \cdot \dfrac{1+\sqrt{5}}{2} + b \cdot \dfrac{1-\sqrt{5}}{2} - 1 \end{cases} \quad a = \frac{5+2\sqrt{5}}{5} , \ b = \frac{5-2\sqrt{5}}{5} \Rightarrow f(h) = \left(\frac{5+2\sqrt{5}}{5}\right)\left(\frac{1+\sqrt{5}}{2}\right)^h + \left(\frac{5-2\sqrt{5}}{5}\right)\left(\frac{1-\sqrt{5}}{2}\right)^h - 1$$

max : $2^{h+1} - 1$, min : $\left(\dfrac{5+2\sqrt{5}}{5}\right)\left(\dfrac{1+\sqrt{5}}{2}\right)^h + \left(\dfrac{5-2\sqrt{5}}{5}\right)\left(\dfrac{1-\sqrt{5}}{2}\right)^h - 1$

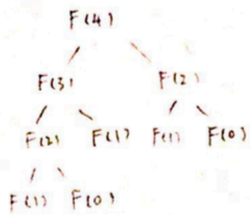
DE of max : $2^{h+1} - 1 = (2^{h-1+1} - 1) \cdot 2 + 1 \Rightarrow X_h = 2X_{h-1} + 1$, $X(0) = 1$

DE of min : $Y_h = Y_{h-1} + Y_{h-2} + 1$, $Y(0) = 1$

Boundary : $\left(\dfrac{5+2\sqrt{5}}{5}\right)\left(\dfrac{1+\sqrt{5}}{2}\right)^h + \left(\dfrac{5-2\sqrt{5}}{5}\right)\left(\dfrac{1-\sqrt{5}}{2}\right)^h - 1 \leq$ number of vertices $\leq 2^{h+1} - 1$

6. Fibonacci : $f_n = f_{n-1} + f_{n-2} \Rightarrow c_1(\frac{1+\sqrt{5}}{2})^n + c_2(\frac{1-\sqrt{5}}{2})^n$

$\Rightarrow T(n) = O((\frac{1+\sqrt{5}}{2})^n + (\frac{1-\sqrt{5}}{2})^n) = O((\frac{1+\sqrt{5}}{2})^n) = O(1.618)^n$ (time complexity)



use the stack for Fibonacci recurssive call we will at most use space n for F(n), so space complexity is O(n)

Ogh : put two sticks together to get next number.

$f_n = f_{n-1} + f_{n-2} + \underbrace{f_{n-2}}_{\text{curve a new stick}} = f_{n-1} + 2f_{n-2} \Rightarrow c_1 \cdot 2^n + c_2 \cdot (-1)^n$

count the number of notches on sticks

$\Rightarrow T(n) = O(2^n + (-1)^n) = O(2^n)$ (time complexity)

Because we need to curve a new stick every time, so space complexity is O(n)
However, the time used to curve notches and read notches is much more than time used to calculate by Fibonacci algorithm.
Besides, the space unit for sticks is physically much larger than the unit in the stack. What's more, it is not ecological to use so many sticks