CSC236 Assignment 3

1.)

a)

For any natural number n, let T(n) denote maximum number of steps executed by a call to mystery(A), where n = len(A).

If n = 1, then it evaluates the if condition and returns 0, which takes constant time, represented by a constant value a. Otherwise, lines 3-8 execute.

The recursive call in line 3 decrements n to a quarter of n, therefore line 4 takes T(n/4).

The for loop in line 5 executes n / 2 times, therefore line 5-6 takes b(n/2) where b is a constant value.

The recursive call in line 7 decrements n to a quarter of n, therefore line 7 takes T(n/4).

All other instructions in lines 3-8 take constant time, represented by a constant value k.

Putting it all together, we get the following definition for T(n):

$$T(n) = \begin{cases} a, & n = 1 \\ 2T(n/4) + b(n/2) + k, & n \geq 2 \end{cases}$$

b)

Assume $n \geq 2$, then $T(n) = 2T(n/4) + b(n/2) + k$.

By the master theorem, c = 2, d = 4, k = 1.

$\log_d c = \log_4 2 = 1/2 < 1 = k$. Therefore, $T(n) \in O(n^k)$, therefore $T(n) \in O(n)$.

2.)

$f(0) = a$

$f(1) = b$

$f(2) = 2f(1) - f(0) + 1 = 2b - a + 1$

$f(3) = 2f(2) - f(1) + 1 = 3b - 2a + 3$

$f(4) = 2f(3) - f(2) + 1 = 4b - 3a + 6$

$f(n) = nb - (n - 1)a + (n(n - 1))/2$

Therefore, the closed-form expression for $f(n)$ is $f(n) = nb - (n - 1)a + (n(n - 1))/2$.

3.)

a)

The initial deposit into the account is \$1000. Thus, $f(0)$ is equal to \$1000.

For $n \geq 1$ we have

$(0.03/12)f(n - 1) + f(n - 1) + \$200 = (1 + 0.03/12)f(n - 1) + \$200 = bf(n - 1) + \$200$

where $b = 1 + 0.03/12$.

We get the following definition for $f(n)$:

$$f(n) = \begin{cases} 1000, & n = 0 \\ bf(n - 1) + 200, & n \geq 1 \end{cases}$$

b)

$f(0) = 1000$

$f(1) = bf(0) + 200 = 1000b + 200$

$f(2) = bf(1) + 200 = 1000b^2 + 200(1 + b)$

$f(3) = bf(2) + 200 = 1000b^3 + 200(1 + b + b^2)$

$f(n) = 1000b^n + 200\sum_{i=0}^{n-1} b^i$

$f(n) = 1000b^n + 200((1 - b^n)/(1 - b))$

$f(n) = 1000b^n - 80000(1 - b^n)$ # $1 - b = -0.0025$ and $200 / -0.0025 = -80000$

Therefore, the closed-form expression for $f(n)$ is $f(n) = 1000b^n - 80000(1 - b^n)$.

c)

$P(n)$: $f(n) = 1000b^n - 80000(1 - b^n)$.

The goal is to prove for all $n \in N$, $P(n)$.

Base Case: Let $n = 0$.

By definition of f, $f(0) = 1000$.

$1000b^0 - 80000(1 - b^0) = 1000 - 80000(1 - 1) = 1000 - 80000(0) = 1000$.

Therefore, $f(n) = 1000b^n - 80000(1 - b^n)$ for $n = 0$ so $P(0)$ holds.

Induction Step: Let $k \in N$, suppose $P(k)$, i.e., $f(k) = 1000b^k - 80000(1 - b^k)$. [IH]

WTP: $P(k + 1)$, i.e., $f(k + 1) = 1000b^{k+1} - 80000(1 - b^{k+1})$

$f(k + 1) = bf(k) + 200$ # by defintion of f since $k + 1 \geq 1$

$f(k + 1) = b(1000b^k - 80000(1 - b^k)) + 200$ # by IH, $f(k) = 1000b^k - 80000(1 - b^k)$

$f(k + 1) = b(1000b^k - 80000 + 80000b^k) + 200$

$f(k + 1) = 1000b^{k+1} - 80000b + 80000b^{k+1} + 200$

$f(k + 1) = 1000b^{k+1} - 80200 + 80000b^{k+1} + 200$ # $b = 1.0025$ and $-80000 * 1.0025 = -80200$

$f(k + 1) = 1000b^{k+1} - 80000 + 80000b^{k+1}$

$f(k + 1) = 1000b^{k+1} - 80000(1 - b^{k+1})$ # factor out -80000

Therefore, $P(k + 1)$ holds and the proof is complete.