Xi Liu, xl3504, Homework 1

for this entire assignment, I use “IH” as acronym for “inductive hypothesis”

Question 0:
None

Question 1:
let $P(n)$ be the proposition such that

$$(1-r)(1+r+r^2+...+r^{n-1}) = 1-r^n, \qquad \forall n \in \mathbb{N}$$

1.
base step:
$P(1)$ is true, when $n = 1$, $\quad n-1 = 0$, substitute 0 for $n-1$ of left-hand side and substitute 1 for n of right-hand side

$$(1-r)(1) = 1-r^1$$

2.
inductive step:
assume $P(k)$ is true for some positive integer k, or equivalently, assume

$$(1-r)(1+r+r^2+...+r^{k-1}) = 1-r^k$$

is true

$$(1-r)(1+r+r^2+...+r^k) = (1-r)(1+r+r^2+...+r^{k-1}+r^k)$$

$$\begin{aligned} &= (1-r)(1+r+r^2+...+r^{k-1}) + (1-r)r^k \\ &\overset{\text{IH}}{=} 1-r^k+(1-r)r^k \\ &= 1-r^k+r^k-r(r^k) \\ &= 1-r^{k+1} \end{aligned}$$

so $P(k+1)$ is true
by mathematical induction, $P(n)$ is true $\forall n \in \mathbb{N}$

3.
rewrite equation (1) in equivalent form:

$$(1-r)(1+r+r^2+...+r^{n-1}) = 1-r^n$$

$$\sum_{i=0}^{n-1} r^i = 1+r+r^2+...+r^{n-1} = \frac{1-r^n}{1-r}$$

$$\sum_{i=0}^{n} r^i = 1 + r + r^2 + ... + r^n = \frac{1 - r^{n+1}}{1 - r}$$

now evaluate the following sum:

$$2^0 \cdot 3^n + 2^1 \cdot 3^{n-1} + 2^2 \cdot 3^{n-2} + ... + 2^n \cdot 3^{n-n} = \sum_{i=0}^{n} \left(2^i \cdot 3^{n-i}\right)$$

$$= \sum_{i=0}^{n} \left(2^i \left(\frac{3^n}{3^i}\right)\right)$$
$$= 3^n \sum_{i=0}^{n} \left(\frac{2^i}{3^i}\right)$$
$$= 3^n \sum_{i=0}^{n} \left(\frac{2}{3}\right)^i$$

apply equation (1) with $r = \frac{2}{3}$

$$= 3^n \left(\frac{1 - (2/3)^{n+1}}{1 - (2/3)}\right)$$
$$= 3^n \left(\frac{1 - (2/3)^{n+1}}{1/3}\right)$$
$$= 3 \cdot 3^n \left(1 - (2/3)^{n+1}\right)$$
$$= 3^{n+1} \left(1 - (2/3)^{n+1}\right)$$
$$= 3^{n+1} - 2^{n+1}$$

Question 2:
1.
false
let $f(n) = n$, $g(n) = n^2$
$f(n) = O(g(n))$ or equivalently $n = O(n^2)$ because there exist positive constants c and n_0 such that $0 \le n \le cn^2$ for all $n \ge n_0$

$$n \le cn^2$$

dividing by n yields

$$1 \le cn$$

we can make the inequality hold for any value of $n \ge 1$ by choosing any constant $c \ge 1$

claim: but $g(n) \ne O(f(n))$ because $n^2 \ne O(n)$
proof: for contradiction, assume $n^2 = O(n)$, then there exist positive constants c and n_0 such that $0 \le n^2 \le cn$ for all $n \ge n_0$

$$n^2 \le cn$$

dividing by n yields

$$n \le c$$

which cannot remain true for arbitrary large n, since c is a constant

2.
true
$f = O(g)$ means $f(n) \le c_1 g(n)$ for all $n \ge n_1$, when c_1 and n_1 are some positive constants
$g = O(h)$ means $g(n) \le c_2 h(n)$ for all $n \ge n_2$, when c_2 and n_2 are some positive constants
when $n \ge n_1$ and $n \ge n_2$, multiply $g(n) \le c_2 h(n)$ by c_1 yields

$$c_1 g(n) \le c_1 c_2 h(n)$$

so

$$f(n) \le c_1 g(n) \le c_1 c_2 h(n)$$

let $c_3 = c_1 c_2$, then

$$f(n) \le c_3 h(n)$$

so

$$f(n) = O(h(n))$$

3.
false
let $f(n) = n^2 + n$, $g(n) = n^2$, then $\forall n \ge 1,\ f(n) > g(n)$
$f(n) = O(g(n))$ or equivalently $n^2 + n = O(n^2)$ because there exist positive constants c and n_0 such that $0 \le n^2 + n \le cn^2$ for all $n \ge n_0$
$n^2 + n \le cn^2$ can remain true when $c \ge 2$ and $n \ge 1$
$g(n) = O(f(n))$ or equivalently $n^2 = O(n^2 + n)$ because there exist positive constants c and n_0 such that $0 \le n^2 \le c(n^2 + n)$ for all $n \ge n_0$
$n^2 \le c(n^2 + n)$ can remain true when $c \ge 1$ and $n \ge 1$

$$f(n) - g(n) = n^2 + n - n^2 = n \ne O(1)$$

proof of $n \ne O(1)$: for contradiction, assume $n = O(1)$, then there exist positive constants c and n_0 such that $0 \le n \le c \cdot 1$ for all $n \ge n_0$

$$n \le c$$

which cannot remain true for arbitrary large n, since c is a constant
so $f(n) - g(n) = n \ne O(1)$

4.
true
$f = O(g)$ means $f(n) \le c_1 g(n)$ for all $n \ge n_1$, when c_1 and n_1 are some positive constants
$g = O(f)$ means $g(n) \le c_2 f(n)$ for all $n \ge n_2$, when c_2 and n_2 are some positive constants

multiply $g(n) \leq c_2 f(n)$ by c_1 yields

$$c_1 g(n) \leq c_1 c_2 f(n)$$

so

$$f(n) \leq c_1 g(n) \leq c_1 c_2 f(n)$$

divide by c_1 yields

$$\frac{f(n)}{c_1} \leq g(n) \leq c_2 f(n)$$

let $f(n)$ be the function that is divided by each of the items in the above inequality, then

$$\frac{f(n)}{f(n)/c_1} \geq \frac{f(n)}{g(n)} \geq \frac{f(n)}{c_2 f(n)}$$

$$c_1 \geq \frac{f(n)}{g(n)} \geq \frac{1}{c_2}$$

since $f(n)/g(n) \leq c_1 \cdot 1$
$f/g = O(1)$

5.
false
let $h(n) = n$, $f(n) = n^2$, $g(n) = n^3$
$f = O(g)$ or equivalently $n^2 = O(n^3)$, since $n^2 \leq c_1 n^3$ for all $n \geq n_1$, when $c_1 \geq 1$ and $n_1 \geq 1$
$h = O(g)$ or equivalently $n = O(n^3)$, since $n \leq c_2 n^3$ for all $n \geq n_2$, when $c_2 \geq 1$ and $n_2 \geq 1$
but $f \neq O(h)$ or equivalently $n^2 \neq O(n)$, since $n^2 \not\leq c_3 n$ or equivalently $n^2 > c_3 n$ for all $n \geq n_3$, when $c_3 \geq 1$ and $n_3 > c_3$

Question 3:
the order is:
$\sqrt{n}\log_2 n,\ 2^{\log_3 n},\ n^2,\ 2^n,\ n!$

Question 4:
let $P(n)$ be the proposition that $f_n > 3n$ for all $n > 9$

base step:
$P(10)$ is true, because $f_{10} = f_9+f_8+f_7 = 24+13+7 = 44 > 3n = 3(10) = 30$
$P(11)$ is true, because $f_{11} = f_{10} + f_9 + f_8 = 44 + 24 + 13 = 81 > 3(11) = 33$
$P(12)$ is true, because $f_{12} = f_{11}+f_{10}+f_9 = 81+44+24 = 149 > 3(12) = 36$

inductive step:
for the inductive hypothesis, assume $P(k)$ is true for an arbitrary integer $k > 9$; that is, assume that $f_k = f_{k-1} + f_{k-2} + f_{k-3} > 3k$

$$f_{k+1} = f_{(k+1)-1} + f_{(k+1)-2} + f_{(k+1)-3} = f_k + f_{k-1} + f_{k-2}$$

because of the inductive hypothesis, $f_k > 3k$
smallest possible f_{k-1} is when $k = 10$, then $f_{k-1} = f_{10-1} = f_9 = 24$
smallest possible f_{k-2} is when $k = 10$, then $f_{k-2} = f_{10-2} = f_8 = 13$
so

$$f_{k+1} = f_k + f_{k-1} + f_{k-2} > 3k + 24 + 13 > 3k + 3 = 3(k + 1)$$

so $P(k + 1)$ is true
by mathematical induction $P(n)$ is true for all $n > 9$

Question 5:
1.
transpositions of array $(7, 5, 2, 6, 9)$:
$(1, 2),\ (1, 3),\ (1, 4),\ (2, 3)$

2.
smallest number of transpositions happen when array is sorted in ascending order: $(1, 2, ..., n)$, smallest number of transpositions = 0

largest number of transpositions happen when array is sorted in descending order: $(n, n-1, n-2, ..., 1)$, there will be $n-i$ transpositions with the ith index, then

$$\text{largest number of transpositions} = (n-1) + (n-2) + ... + 1$$

$$= \sum_{i=1}^{n-1}(n-i) = \sum_{i=1}^{n-1} i = \frac{(n-1)((n-1)+1)}{2} = \frac{n(n-1)}{2}$$

3.

```
int merge(int A[], int p, int q, int r)
{
    int left_len = q - p + 1;
    int right_len = r - q;
    int L[left_len + 1];
    int R[right_len + 1];
    L[left_len + 1] = ∞;
    R[right_len + 1] = ∞;
    for(int i = 1; i < left_len; i++)
        L[i] = A[p + i];
    for(int i = 1; i < right_len; i++)
        R[i] = A[(q + 1) + i];
```

```
    int i = 1;
    int j = 1;
    int counter = 0;
    for(int k = p; k < r; k++)
    {
        if(L[i] <= right[r_i])
        {
            A[k] = L[i];
            i = i + 1;
        }
        else
        {
            A[k] = R[j];
            j = j + 1;
            counter = counter + left_len - i + 1;
        }
    }
    return counter;
}

int merge_transposition(int A[], int p, int r)
{
    int c = 0;
    if(p < r)
    {
        int q = p + ⌊ (r - p) / 2 ⌋;
        int left = merge_transposition(A, p, q);
        int right = merge_transposition(A, q + 1, r);
        c = left + right + merge(A, p, q, r);
    }
    return c;
}
```

correctness of merge

loop invariant:

at the start of the each for iteration (the loop that rewrites original array A using elements in L and R arrays), $A[p...k-1]$ contains $final\_index - initial\_index + 1 = (k-1) - p + 1 = k - p$ smallest elements of L and R, in sorted order, *counter* stores the number of transpositions $(a,\ b)$ such that $p \leq a < p + i - 1$ and $q + 1 \leq b \leq j$

Moreover, $L[i]$ and $R[j]$ are smallest elements of their arrays that have not been copied back to A

initialization:

before first iteration of the loop, $k = p = 1$, so subarray $A[p...k-1]$ is empty. This empty subarray contains $k - p = 0$ smallest elements of L and R. Since $i = j = 1$, both $L[i]$ and $R[j]$ are the smallest elements of their arrays that have not been copied back to A, $i = 1$, $p \leq a < p + 1 - 1$, since $a \geq p$ and $a < p$ is a contradiction, there is no such number a so there is no transposition and *counter* is 0

maintenance: $P(k) \to P(k+1)$

if $L[i] > R[j]$, then $R[j]$ is the smallest element that has not yet copied back into A,

based on the inductive hypothesis (loop invariant), $A[p...k-1]$ contains $k-p$ smallest elements of array L and R, and *counter* stores the number of transpositions $(a,\ b)$ such that $p \leq a < p + i - 1$ and $q + 1 \leq b \leq j \quad (P(k))$.

after line 14 copies $L[i]$ into $A[k]$, the subarray $A[p...k]$ contains $final\_index - initial\_index + 1 = k - p + 1$ smallest elements of array L and R. since the assumption is $L[i] > R[j]$, so $R[j]$ is less than left_len - i + 1 elements of $L[i...left\_len]$, but $i < j$, so there are left_len - i + 1 transpositions associated with $L[i...left\_len]$ and $R[j]$, so after adding left_len - i + 1 to *counter* and increment j by 1 and increment k by 1, the loop invariant is maintained for $k + 1 \quad (P(k+1))$.

if $L[i] \leq R[j]$, then merge appropriate action to maintain the loop invariant with the roles of L and R interchanged

termination:

at termination, $k = r + 1$, $i =$ left_len, $j =$ right_len. By the loop invariant, the subarray $A[p...k-1] = A[p...(r+1)-1] = A[p...r]$, contains $final\_index - initial\_index + 1 = r - p + 1$ smallest elements of

$L[1...n_1+1]$ and $R[1...n_2+1]$ in sorted order, *counter* completed the addition of all transpositions $(a,\ b)$ associated with $L[i...\text{left\_len}]$ and $R[j]$ for all $p \le i \le p + \text{left\_len}$, $q+1 \le j \le q + \text{right\_len}$, $p \le a < p+i-1$, and $q+1 \le b \le j$

show $\Theta(n \lg n)$ is the run time bounds:
recurrence is given by
$T(n) = \begin{cases} \Theta(1) & if\ n = 1 \\ T(\lceil n/2 \rceil) + T(\lfloor n/2 \rfloor) + \Theta(n) & if\ n > 1 \end{cases}$
assume $T(n) \le c(n-2)\lg(n-2)$ is true for all positive $m < n$, in particular for $m = \lceil n/2 \rceil$ and $m = \lfloor n/2 \rfloor$, or equivalently assume $T(\lceil n/2 \rceil) \le c(\lceil n/2 \rceil - 2)\lg(\lceil n/2 \rceil - 2)$ and $T(\lfloor n/2 \rfloor) \le c(\lfloor n/2 \rfloor - 2)\lg(\lfloor n/2 \rfloor - 2)$
assume $\Theta(n) = kn$ for the recurrence stated above, where k is a positive constant

$$\begin{aligned}
T(n) &= T(\lceil n/2 \rceil) + T(\lfloor n/2 \rfloor) + kn \\
&\le c(\lceil n/2 \rceil - 2)\lg(\lceil n/2 \rceil - 2) + c(\lfloor n/2 \rfloor - 2)\lg(\lfloor n/2 \rfloor - 2) + kn \\
&\le c(n/2+1-2)\lg(n/2+1-2) + c(n/2+1-2)\lg(n/2+1-2) + kn \\
&= 2c(n/2+1-2)\lg(n/2+1-2) + kn \\
&= 2c(n/2-1)\lg(n/2-1) + kn \\
&= 2c((n-2)/2)\lg((n-2)/2) + kn \\
&= c(n-2)\lg((n-2)/2) + kn \\
&= c(n-2)(\lg(n-2) - \lg 2) + kn \\
&= c(n-2)(\lg(n-2) - 1) + kn \\
&= c(n-2)\lg(n-2) - c(n-2) + kn \\
&= c(n-2)\lg(n-2) + n(k-c) + 2c \\
&= c(n-2)\lg(n-2) - (n(c-k) - 2c) \\
&\le c(n-2)\lg(n-2) \qquad if\ n(c-k) - 2c \ge 0
\end{aligned}$$

$$n(c-k) - 2c \ge 0$$
$$n(c-k) \ge 2c$$

$$n \geq \frac{2c}{c-k}$$

$$\text{so, pick} \quad n \geq n_0 = \frac{2c}{c-k} \quad \text{and} \quad c > k$$

so

$$T(n) = O(n \lg n)$$

need to show $T(n) \geq cn \lg n$ for all $n \geq n_0$, where c and n_0 are positive constants

assume $T(n) \geq c(n+2)\lg(n+2)$ is true for all positive $m < n$, in particular for $m = \lceil n/2 \rceil$ and $m = \lfloor n/2 \rfloor$, or equivalently assume $T(\lceil n/2 \rceil) \geq c(\lceil n/2 \rceil + 2)\lg(\lceil n/2 \rceil + 2)$ and $T(\lfloor n/2 \rfloor) \geq c(\lfloor n/2 \rfloor + 2)\lg(\lfloor n/2 \rfloor + 2)$

$$\begin{aligned}
T(n) &= T(\lceil n/2 \rceil) + T(\lfloor n/2 \rfloor) + kn \\
&\geq c(\lceil n/2 \rceil + 2)\lg(\lceil n/2 \rceil + 2) + c(\lfloor n/2 \rfloor + 2)\lg(\lfloor n/2 \rfloor + 2) + kn \\
&\geq c(n/2 - 1 + 2)\lg(n/2 - 1 + 2) + c(n/2 - 1 + 2)\lg(n/2 - 1 + 2) + kn \\
&\geq 2c(n/2 - 1 + 2)\lg(n/2 - 1 + 2) + kn \\
&= 2c(n/2 + 1)\lg(n/2 + 1) + kn \\
&= 2c((n+2)/2)\lg((n+2)/2) + kn \\
&= c(n+2)(\lg(n+2) - \lg 2) + kn \\
&= c(n+2)\lg(n+2) - c(n+2) + kn \\
&= c(n+2)\lg(n+2) + n(k-c) - 2c \\
&\geq c(n+2)\lg(n+2) \\
&if\ n(k-c) - 2c \geq 0, \quad i.e. \quad n \geq \frac{2c}{k-c}
\end{aligned}$$

$$\text{so, pick} \quad n \geq n_0 = \frac{2c}{k-c} \quad \text{and} \quad k > c$$

so

$$T(n) = \Omega(n \lg n)$$

because $T(n) = O(n \lg n)$ and $T(n) = \Omega(n \lg n)$, so $T(n) = \Theta(n \lg n)$

Question 6:
the inductive step is false for $n = 2$
if the set S has 2 horses and $S = \{A,\ B\}$,
if exclude A, and look at the set $\{B\}$, all of the remaining element(s) have the same color because there is only 1 element in $\{B\}$
if exclude B, and look at the set $\{A\}$, all of the remaining element(s) have the same color because there is only 1 element in $\{A\}$
but this does not show that A and B must have the same color

Honors Question 1:

$$\tilde{O}(g(n)) = \{f(n) : \text{there exist positive constants } c \text{ and } n_0 \text{ such that}$$
$$0 \leq f(n) \leq \log_2 n \cdot g(n),\ \forall n \geq n_0\}$$

this new notion have transitivity:

$$f(n) = \tilde{O}(g(n)) \text{ and } g(n) = \tilde{O}(h(n)) \text{ imply } f(n) = \tilde{O}(h(n))$$

Honors Question 2:
let $P(n)$ be the proposition that a convex n-gon has $n(n-3)/2$ diagonals

base step:
$P(3)$ is true, because the convex 3-gon is a triangle, it has $3(3-3)/2 = 0$ diagonals

inductive step:
assume $P(k)$ is true, or equivalently assume a convex k-gon has $k(k-3)/2$ diagonals
if 1 vertex is added to the convex $k-$gon, then $(k+1)-2$ new lines can be drawn, so the number of diagonals in the newly formed convex $k+1$-gon is

$$\text{number of diagonals in convex } k\text{-gon} + (k+1) - 2$$

$$\begin{aligned}
&= k(k-3)/2 + (k+1) - 2 \\
&= (k^2 - 3k)/2 + k - 1 \\
&= k^2/2 - 3k/2 + k - 1 \\
&= k^2/2 - k/2 - 1 \\
&= \frac{1}{2}(k^2 - k - 2) \\
&= \frac{1}{2}(k-2)(k+1) \\
&= (k+1)((k+1)-3)/2
\end{aligned}$$

so $P(k+1)$ is true
by mathematical induction $P(n)$ is true for all $n \geq 3$