Throughout these questions the notation $\mathbb{N}^*$ will be used as a shorthand for $\mathbb{N} \cup \{0\}$.

1. Let $a_n = 2n + 3 \quad \forall n \in \mathbb{N}$. Compute the following.

(a) $\sum_{n=1}^{4} a_n \quad = (2+3) + (4+3) + (6+3) + (8+3) = \mathbf{32.}$

(b) $\prod_{n=1}^{4} a_n \quad = (2+3)(4+3)(6+3)(8+3) = \mathbf{3465.}$

2. Let $(x_n)_{n\in\mathbb{N}^*} \subseteq \mathbb{Q}$ be such that $\quad x_0 = 1 \quad$ and $\quad x_{n+1} = \dfrac{x_n}{2} \quad \forall n \in \mathbb{N}^*$.

Prove by mathematical induction that $x_n = 2^{-n} \quad \forall n \in \mathbb{N}^*$.

Basis step: **For $n = 0$ formula gives $x_0 = 2^{-0} = 1$, agreeing with the definition.**

Inductive step: **Assume the formula is correct up to and including some fixed $n \in \mathbb{N}^*$. Then**

$$\begin{aligned} x_{n+1} &= \frac{x_n}{2} && \textbf{(from the implicit definition)} \\ &= \frac{2^{-n}}{2} && \textbf{(by the inductive assumption)} \\ &= 2^{-(n+1)} \end{aligned}$$

and so the formula is also correct for $n+1$.

3. Let $x \in \mathbb{Q}$. Prove that $\quad \forall N \in \mathbb{N}^* \quad (1 + x) \sum_{n=0}^{N} (-x)^n = 1 - (-x)^{N+1}$.

Use mathematical induction.

Basis step: **For $N = 0$:**

$$\begin{aligned} \textbf{LHS} &= (1+x)(-x)^0 = 1+x. \\ \textbf{RHS} &= 1 - (-x)^{0+1} = 1+x \quad = \quad \textbf{LHS.} \end{aligned}$$

Inductive step: **Assume the formula is correct up to and including some fixed $n \in \mathbb{N}^*$. Then for $N+1$**

$$\begin{aligned} \textbf{LHS} = (1+x)\sum_{n=0}^{N+1}(-x)^n &= (1+x)\left[\sum_{n=0}^{N}(-x)^n \; + \; (-x)^{N+1}\right] \\ &= 1 - (-x)^{N+1} + (1+x)(-x)^{N+1} \quad \textbf{(by the in.ass.)} \\ &= 1 - (-x)^{N+1} + (-x)^{N+1} + x(-x)^{N+1} \\ &= 1 + x(-x)^{N+1} \; = \; 1 - (-x)^{N+2} \; = \; \textbf{RHS for } N+1. \end{aligned}$$

and so the formula is also correct for $N+1$.

4. (Population dynamics: Hassel's model). Let $(p_n)_{n\in\mathbb{N}^*} \subseteq \mathbb{Q}$ be the size (in hundreds of individuals) of a population of ponies at time check n. Assume that $p_0 = x$ for some x with $0 < x < 1$ and that $\forall n \in \mathbb{N}^* \;\; p_{n+1} = \dfrac{2p_n}{1 + p_n}$. Prove that $p_n \leq 1 \quad \forall n \in \mathbb{N}$.

Basis step: **For $n = 1$:** $\quad p_1 = \dfrac{2p_0}{1+p_0}$ **(by def.)** $= \dfrac{2x}{1+x} < \dfrac{2x}{x+x} = 1$,
since $x < 1$ is given. So the inequality is correct for $n = 1$.

Inductive step: **Assume the inequality holds up to and including some fixed $n \in \mathbb{N}$. Then for $n+1$**

$$p_{n+1} = \frac{2p_n}{1+p_n} \; \textbf{(by def.)} \; < \; \frac{2p_n}{p_n + p_n} \; = \; 1$$

since $p_n < 1$ by the inductive assumption. Thus the inequality also holds for $n+1$.

5. Let $v_n \in \mathbb{Q}$ represent the quantity, at time $n \in \mathbb{N}^\star$, of a certain virus in the bloodstream. Let a_n represent the number of antibodies produced to fight the virus at the same time n. Assume that a_0 and v_0 are positive and that, $\forall n \in \mathbb{N}^\star$, a_n and v_n are related by the equations

$$\text{A:}\quad a_{n+1} = a_n + 2v_n, \qquad\qquad \text{V:}\quad v_{n+1} = 4v_n - a_n.$$

(a) Explain what these equations model.
Equation A says that at each time step the number of antibodies goes up by an amount equal to twice the quantity of virus present at that time. Equation V says that at each time step the quantity of virus quadruples but is then cut back by an amount equal to the number of antibodies present at that time.

(b) Prove that $v_{n+2} = 5v_{n+1} - 6v_n \quad \forall n \in \mathbb{N}^*$.

$$\begin{aligned} v_{n+2} &= 4v_{n+1} - a_{n+1} \quad \textbf{(from V)} \\ &= 4v_{n+1} - (a_n + 2v_n) \quad \textbf{(from A)} \\ &= 4v_{n+1} - ((4v_n - v_{n+1}) + 2v_n) \quad \textbf{(from V)} \\ &= 5v_{n+1} - 6v_n. \end{aligned}$$

(c) Use mathematical induction to prove that $v_n = (2v_0 - a_0)3^n + (a_0 - v_0)2^n \quad \forall n \in \mathbb{N}^*$.

Basis step: **For $n{=}0$, $n{=}1$ formula gives**
$v_0 = (2v_0 - a_0)3^0 + (a_0 - v_0)2^0 = 2v_0 - a_0 + a_0 - v_0 = v_0$, **as it should, and**
$v_1 = (2v_0 - a_0)3^1 + (a_0 - v_0)2^1 = 6v_0 - 3a_0 + 2a_0 - 2v_0 = 4v_0 - a_0$ **agreeing with V.**

Inductive step: **Assume the formula is correct up to and including some fixed $n \in \mathbb{N}$. Then**

$$\begin{aligned} v_{n+1} &= 5v_n - 6v_{n-1} \quad \textbf{(from part (b))} \\ &= 5[(2v_0 - a_0)3^n + (a_0 - v_0)2^n] - 6[(2v_0 - a_0)3^{n-1} + (a_0 - v_0)2^{n-1}] \quad \textbf{(by in. ass.)} \\ &= 5(2v_0 - a_0)3^n - 2(2v_0 - a_0)3^n + 5(a_0 - v_0)2^n - 3(a_0 - v_0)2^n \\ &= 3(2v_0 - a_0)3^n + 2(a_0 - v_0)2^n \;=\; (2v_0 - a_0)3^{n+1} + (a_0 - v_0)2^{n+1} \end{aligned}$$

and so the formula is also correct for $n{+}1$.

(d) Under what initial conditions does the model predict that the virus will eventually be eliminated?
The coefficient $(2v_0 - a_0)$ of 3^n will need to be negative. So the condition is $a_0 > 2v_0$; *i.e.* the initial number of antibodies needs to be more than twice the initial quantity of virus.

6. Compute the following sums:

(a) $\displaystyle\sum_{n=1}^{30} n \quad = 30(\tfrac{1+30}{2}) = \mathbf{465.}$

(b) $\displaystyle\lfloor \sum_{n=1}^{30} ((\tfrac{5}{4})\, n + 3) \rfloor \quad = \lfloor \tfrac{5}{4} \sum_{n=1}^{30} n + \sum_{n=1}^{30} 3 \rfloor = \lfloor \tfrac{5}{4}(465) + 30(3) \rfloor = \lfloor 581\tfrac{1}{4} + 90 \rfloor = \mathbf{671}.$

(c) $\displaystyle\lfloor \sum_{n=1}^{30} (\tfrac{5}{4})^n \rfloor \quad = \lfloor \tfrac{5}{4}((\tfrac{5}{4})^{30} - 1)/(\tfrac{5}{4} - 1) \rfloor = \lfloor 5(807.7936 - 1) \rfloor = \lfloor 4033.968 \rfloor = \mathbf{4033}.$

(d) $\displaystyle\lfloor \sum_{n=1}^{30} ((\tfrac{5}{4})^n + 3) \rfloor \quad = \lfloor \sum_{n=1}^{30} (\tfrac{5}{4})^n + \sum_{n=1}^{30} 3 \rfloor = \lfloor 4033.968 + 90 \rfloor = \mathbf{4123}.$

7. Selection sort is used to sort the letters of the word SELECTION into alphabetical order. This requires eight applications of the Least element algorithm. Write out the state of the word after each of these eight applications.

CELESTION $\longrightarrow$ **CELESTION** $\longrightarrow$ **CEELSTION** $\longrightarrow$ **CEEISTLON**
$\longrightarrow$ **CEEILTSON** $\longrightarrow$ **CEEILNSOT** $\longrightarrow$ **CEEILNOST** $\longrightarrow$ **CEEILNOST**

8. Merge sort is used to sort the letters of the word TRANSFORMATIONAL into alphabetical order. This requires fifteen applications of the Merge algorithm, spread over four iterations of the Merge sort algorithm. Write out the state of the word after each of these four iterations.

RT AN FS OR AM IT NO AL
$\longrightarrow$ **ANRT FORS AIMT ALNO**
$\longrightarrow$ **AFNORRST AAILMNOT**
$\longrightarrow$ **AAAFILMNNOORRSTT**

9. The list $(1, 3, 5, 7, 8, 6, 4, 2)$ is to be sorted into ascending order.

(a) How many comparisons will be required using the Selection sort algorithm?

$7 + 6 + 5 + 4 + 3 + 2 + 1 = 7(\frac{7+1}{2}) = \mathbf{28}$.

(b) Is the number of comparisons required by Selection sort affected by the order of the original list? If so, what is the most that would ever be required (for list length 8)?

Not affected.

(c) How many comparisons will be required using the Merge sort algorithm?

13 57 68 24; 4 **comparisons.**
1357 2468; $2 + 2 = 4$ **comparisons.**
12345678; 7 **comparisons**
Total: 15 **comparisons**

(d) Is the number of comparisons required by Merge sort affected by the order of the original list? If so, what is the most that would ever be required (for list length 8)?

Yes it is affected. Since $8 = 2^3$**, an upper bound is** $T_3 = 3(2^3) = \mathbf{24}$.
However the actual maximum is 17. See Q10b.

10. Merge sort is to be used to sort a list of 8 distinct numbers from the set $\{1, 2, 3, 4, 5, 6, 7, 8\}$ into ascending order. Give examples of list orders that give the algorithm the

(a) least trouble; (best case complexity) $(1, 2, 3, 4, 5, 6, 7, 8)$ $4+4+4 = 12$ **comparisons**

(b) most trouble. (worst case complexity) $(1, 5, 3, 7, 8, 4, 6, 2)$ $4+6+7 = 17$ **comparisons**

"Trouble" is to be measured by the number of comparisons required. Calculate this number in each case.

(c) What should we mean by an average amount of trouble (average case complexity), and how could it be calculated? [Do not attempt to actually do the calculation, unless you have a lot of spare time to kill.]

There are $8! = 40320$ **different orderings of** $\{1, 2, 3, 4, 5, 6, 7, 8\}$**. We would have to calculate the number of comparisons required to sort each of these into order and then take the average.**