Instructions: *See Worksheets 1 and 2*

The first question is really just to check you understand some relevant notation. The next three questions are mostly about the proof technique known as mathematical induction, which is an essential tool in the development of mathematics. The proofs you are asked to try relate to special cases of general formulae for arithmetic, geometric and mixed sequences covered in lectures. The last two questions concern sorting algorithms. They offer you the chance to practice following the 'instructions' of an algorithm, and also to analyse its efficiency.

Question 1

(a) Evaluate: (i) $\sum_{k=1}^{5}(k-1)^2$ (ii) $\prod_{k=1}^{5}\left(\frac{k+1}{k}\right)$

(b) Calculate a_6 given that

$$a_0 = 2, \quad a_1 = 3, \quad \text{and} \quad \forall n \in \mathbb{N} \quad a_{n+1} = a_n(a_{n-1} - 1).$$

Question 2

A sequence $(a_n)_{n\in\{0,\ldots,N\}}$ defined implicitly by

$$a_0 = a, \qquad \forall n \in \mathbb{N} \quad n \leq N \Rightarrow a_n = a_{n-1} + d$$

is called *arithmetic*. By supplying the missing bits (a)–(h), complete the proof below that, for an arithmetic sequence, every a_n is given by the formula $a_n = a + nd$.

Basis Step: For $n = 0$ the formula gives . . . (a) . . . $= a + 0d = a$ which agrees with the . . . (b) . . . of (a_n).

Inductive Step Assume the formula holds up to and including some fixed . . . (c) . . .. Then

$$\begin{aligned} a_{n+1} &= \ldots (d) \ldots \text{ from the } \ldots (e) \ldots \text{ definition} \\ &= \ldots (f) \ldots + d \text{ by the inductive assumption} \\ &= a + (n+1)d. \end{aligned}$$

So the formula also . . . (g) . . . for $n+1$. By mathematical . . . (h) . . . this proves that the formula holds for all n, $0 \leq n \leq N$.

Question 3

(a) Prove by induction that $\forall n \in \mathbb{N} \quad \sum_{k=0}^{n-1} k = \frac{n(n-1)}{2}$.

(b) Using (a), or otherwise, prove that the sum of any arithmetic sequence is equal to its number of terms times the average of its first and last terms. *e.g.*

$$2 + 5 + 8 + 11 + 14 + 17 + 20 + 23 + 26 = 9\left(\frac{2+26}{2}\right) = 9 \times 14 = 126.$$

Question 4 I borrow \$100 000 at an interest rate of %6 per annum and agree to pay back \$1000 per month. Assuming interest compounding monthly, my debt, in dollars, after n months is given implicitly by

$$a_0 = 100\,000 = a \qquad a_n = (1.005)a_{n-1} - 1000 = ra_{n-1} - f$$

say, where $a = 100\,000$, $r = 1.005$ and $f = 1000$.

(a) Prove by mathematical induction that $\quad \forall n \in \mathbb{N} \quad a_n = ar^n - f\sum_{k=0}^{n-1} r^k.$

(b) Given that for $r \neq 1$ and $n \in \mathbb{N}$, $\displaystyle\sum_{k=0}^{n-1} r^k = \frac{r^n - 1}{r - 1}$, how much will I owe in 10 years time?

Question 5 The letters of the word `TROUNCED` form the list $(X_i)_{1..8}$ = (T,R,O,U,N,C,E,D). This list is to be sorted into alphabetical order using Selection sort. The sorting is to be achieved by progressively modifying an index function π, rather than by shuffling members of the list itself. So initially

$$(X_i)_{1..8} = (X_{\pi(i)})_{1..8} \text{ where } \pi = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \\ 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \end{pmatrix}$$

and when sorting is complete π is sufficiently changed so that $(X_{\pi(i)})_{1..8}$ is in order.

(a) First apply the Least Element algorithm to $(X_i)_{1..8}$. Demonstrate the application by completing the trace table at right.

i	2	3	4	5	6	7	8	9
m	1	2	3	3				
$x_{\pi(i)}$	R	O	U	N				-
$x_{\pi(m)}$	T	R	O	O				

(b) Write out the modified index function π resulting from (a).

(c) Now apply the Least Element algorithm to $(X_{\pi(i)})_{2..8}$ using this modified π, again demonstrating the application by a trace table.

(d) Write out the newly modified index function π resulting from (c).

(e) Without making trace tables, write out the state of index function π after each of the remaining applications of the Least element algorithm needed to complete the Selection sort of (T,R,O,U,N,C,E,D).

(f) What is the total number of comparisons used during this sort?

(g) By contrast, how many comparisons, in total, would be used to sort (T,R,O,U,N,C,E,D) using the Merge sort algorithm? To find out, carry out the Merge sort algorithm on (T,R,O,U,N,C,E,D) and carefully count the comparisons, remembering that when the Merge algorithm reaches a stage where one of its input lists is empty, it does not need any more comparisons to complete its task.

Question 6 In lectures we saw how use the Merge sort algorithm to sort a sequence of length $n = 2^r$ into ascending order. In fact the algorithm can be applied to sequences of any length $n \in \mathbb{N}$. At each iteration the current sorted sub-sequences are merged in pairs as for the 2^r case but if there are an odd number of sub-sequences then the 'left over' one just joins, unchanged, the newly formed sub-)sequences at the next iteration. This will mean that the merge algorithm will sometimes need to merge sequences of unequal lengths, but this causes no problems.

For example, if Merge sort is used to sort the letters of the word `PROVISIONAL` into alphabetical order then the subsequences at each stage will be:

after 0th iteration	`(P),(R),(O),(V),(I),(S),(I),(O),(N),(A),(L);`
after 1st iteration	`(P,R),(O,V),(I,S),(I,O),(A,N),(L);`
after 2nd iteration	`(O,P,R,V),(I,I,O,S),(A,L,N);`
after 3rd iteration	`(I,I,O,O,P,R,S,V),(A,L,N);`
after 4th iteration	`(A,I,I,L,N,O,O,P,R,S,V).`

(a) Apply the Merge sort algorithm to sort the letters of the word `APPROPRIATION` into alphabetical order, showing the results of each iteration as in the example above.

(b) How many comparisons are used to merge sort `APPROPRIATION`? As in Q5(g), remember that when the merge algorithm reaches the stage where one of its input lists is empty, it does not need any more comparisons to complete its task. For example, for `PROVISIONAL` there are only 5 comparisons during the first iteration, 8 in the 2nd, 7 in the 3rd and 5 in the last.

(c) How many comparisons would be used if `APPROPRIATION` were sorted using the Selection sort algorithm?