1. Find the value of \( \sum_{k=1}^{n} (2k-1) \) (5 points)

2. Evaluate the sum \( \sum_{k=0}^{\infty} (4k+1)2^k \) (10 points)

3. Prove \( \log \prod_{k=1}^{n} a_k = \sum_{k=1}^{n} \log(a_k) \) (5 points)

4. (20 points) The following pseudo-code is for a procedure called Insertion Sort, which takes as parameter an array A[1…n] of length n and sorts the sequence (array A) in ascending order.

\[
\text{for } j \leftarrow 2 \text{ to } \text{length}[A] \\
\quad \text{do } key \leftarrow A[j] \\
\quad \quad i \leftarrow j - 1 \\
\quad \quad \text{while } i > 0 \text{ and } A[i] > key \\
\quad \quad \quad \text{do } A[i+1] \leftarrow A[i] \\
\quad \quad \quad \quad i \leftarrow i - 1 \\
\quad \quad \quad A[i+1] \leftarrow key
\]

Rewrite the above procedure to sort the values in the array A in descending order.

5. (20 points) Consider the problem of adding two n-bit integers, sorted in two n-element arrays A and B. The sum of the two integers should be stored in binary form in an (n+1)-element array C. State the problem formally and write pseudo-code for adding the two integers.

6. (20 points) Consider sorting n numbers stored in array A by first finding the smallest element of A and exchanging it with the element in A[1]. Then find the second smallest element of A and exchange it with A[2]. Continue in this manner for the first \( n - 1 \) elements of A. Write pseudo-code for this algorithm, which is known as Selection sort.
7. (20 points) Consider the searching problem:

**Input:** A sequence of \( n \) numbers \( A = \{a_1, a_2, \ldots, a_n\} \) and a value \( v \).

**Output:** An index \( i \) such that \( v = A[i] \) or the special value NIL if \( v \) does not appear in \( A \).

Assume that the sequence \( A \) is sorted. The objective is to search the sequence for the given value \( v \). In order to achieve that, we can check the midpoint of the sequence against \( v \) and eliminate half of the sequence from further consideration. **Binary Search** is an algorithm that repeats this procedure, halving the size of the remaining portion of the sequence each time. Write pseudo-code, either iterative or recursive, for the binary search.