Problem 1 (Hoare logic)

Consider the program P:

Algorithm 1
Input: Two integers $M > 0$ and $N \geq 0$
Output: An integer $k$ such that $k = M^N$

\begin{itemize}
  \item $a \leftarrow M$
  \item $b \leftarrow N$
  \item $k \leftarrow 1$
  \item while $b > 0$ do
    \begin{itemize}
      \item if even($b$) then
        \begin{itemize}
          \item $a \leftarrow a \times a$
          \item $b \leftarrow b/2$
        \end{itemize}
      \item else
        \begin{itemize}
          \item $k \leftarrow k \times a$
          \item $b \leftarrow b - 1$
        \end{itemize}
    \end{itemize}
  \end{itemize}

Note that $/$ is an integer division. For instance, the predicate even($b$) can be implemented as $b = 2 \times (b/2)$.

Find an invariant that is strong enough to prove the correctness of the program.

Use Hoare logic to prove \{(M > 0) \land (N \geq 0)\} P \{k = M^N\}

Problem 2 (Circuits)

1. Let NAND be a gate with the following truth table:

<table>
<thead>
<tr>
<th>Input1</th>
<th>Input2</th>
<th>Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
</tbody>
</table>

Construct the NOT, AND, OR, XOR gates using only the NAND gate.

2. Is it possible to construct NOT, OR, XOR using only the AND gate?
Problem 3 (λ-calculus)

In this problem, we will encode arithmetic on natural numbers in the λ-calculus. The numbers themselves will be represented as follows:

\[ \lambda s.\lambda z.z \] is zero,
\[ \lambda s.\lambda z.s \, z \] is one,
\[ \lambda s.\lambda z.s \, (s \, z) \] is two,
\[ \lambda s.\lambda z.s \, (s \, (s \, z)) \] is three,
\[ \ldots \]

1. Define the successor function in λ-calculus, that is, a function that given a representation of \( n \), returns a representation of \( n + 1 \).

2. Define the addition function.