Q1.
\[ x := y - z; \]
\[ \text{if}(x < 5) \text{ then} \]
\[ x := y \times z; \]
\[ \text{else} \]
\[ x := y - z; \]

**Solution:**

We can calculate \( x := y - z; \) and \( x := y \times z; \) with \( Q \equiv \{ x > 0 \} \) first, we obtain:

1. \( \{ y \times z > 0 \} \ x := y \times z \ \{ x > 0 \} \)
2. \( \{ y - z > 0 \} \ x := y - z; \ \{ x > 0 \} \)

Now we calculate the weakest precondition of the if rule. Based on 1.\( \{ y - z > 0 \} \) and 2.\( \{ y \times z > 0 \}, \) we obtain

3. \( \{ y \times z > 0 \ \land \ x < 5 \} \ \lor \ \{ y - z > 0 \ \land \ x \geq 5 \} \) the if statement \( \{ x > 0 \} \)

Then, we calculate the weakest precondition of the \( x := y - z; \) Based on 3. \( \{ y \times z > 0 \ \land \ x < 5 \} \ \lor \ \{ y - z > 0 \ \land \ x \geq 5 \} \)

4. \( \{ (y \times z > 0 \ \land \ y - z < 5) \ \lor \ (y - z > 0 \ \land \ y - z \geq 5) \} \ x := y - z; \ \{ y \times z > 0 \ \land \ x < 5 \} \ \lor \ \{ y - z > 0 \ \land \ x \geq 5 \} \)

Finally, we have:

5. \( \{ y \times z > 0 \ \land \ y - z < 5 \} \ \lor \ \{ y - z > 0 \ \land \ y - z \geq 5 \} \ \land \ {x > 0} \)

Which equivalent to \( P \equiv \{ y \times z > 0 \ \land \ y - z < 5 \} \ \lor \ \{ y - z > 0 \} \)
Q2.

\[\begin{align*}
i &:= 0; \\
j &:= \text{len(arr)} - 1; \\
\text{while } i \leq j : \\
&\quad \text{if } \text{arr}[i] \leq \text{arr}[0]: \\
&\quad\quad i := i + 1; \\
&\quad \text{else:} \\
&\quad\quad \text{tmp} := \text{arr}[i]; \\
&\quad\quad \text{arr}[i] := \text{arr}[j]; \\
&\quad\quad \text{arr}[j] := \text{tmp}; \\
&\quad\quad j := j - 1;
\end{align*}\]

Solution:

The postcondition is a rigorous restatement of the loop goal, which says all elements \( \text{arr}[0 .. j] \) are less than or equal to \( \text{arr}[0] \), and all elements \( \text{arr}[j + 1 .. \text{len(arr)} - 1] \) are greater than \( \text{arr}[0] \).

\[Q = \forall x \, (0 \leq x \leq j \rightarrow \text{arr}[x] \leq \text{arr}[0]) \land \forall y \, (j < y < \text{len(arr)} \rightarrow \text{arr}[y] > \text{arr}[0])\]

The loop invariant is based on a partial partitioning of the array.

\[P = \forall x \, (0 \leq x < i \rightarrow \text{arr}[x] \leq \text{arr}[0]) \land \forall y \, (j < y < \text{len(arr)} \rightarrow \text{arr}[y] > \text{arr}[0]) \land i \leq j + 1\]

Note that before the first iteration this is satisfied because \( \text{arr}[0] \leq \text{arr}[0] \) and there are no elements in the range \( j+1 .. \text{len(arr)} - 1 \). The loop condition (b) guarantees that \( i \leq j \).

In each iteration one of two things happens:
- \( i \) is incremented when \( \text{arr}[i] \leq \text{arr}[0] \) – this leaves \( i \leq j + 1 \), and all elements up to index \( i \) are \( \leq \text{arr}[0] \)
- \( j \) is decremented when \( \text{arr}[i] > \text{arr}[0] \), but only after swapping the item at \( i \) with the item at \( j \). This leaves \( i \leq j + 1 \), and all elements down to index \( j \) are \( > \text{arr}[0] \)

After termination, we know (by the invariant) that \( i \leq j + 1 \), and by the negation of the loop condition \( (\neg b) \), that \( i > j \). This implies that \( i = j + 1 \).

\[P \land \neg b = \forall x (0 \leq x < i \rightarrow \text{arr}[x] \leq \text{arr}[0]) \land \forall y (j < y < \text{len(arr)} \rightarrow \text{arr}[y] > \text{arr}[0]) \land i = j + 1\]

If we just replace \( i \) with \( j + 1 \) in the quantified portions of the proposition, then we see that \( P \land \neg b \rightarrow Q \) directly.