This quiz has 3 questions, for a total of 30 points.

1. **10 points** Apply the Partition algorithm to the following array. Write down the array after each step, drawing lines between the partitions as was done in the lecture and the textbook.

   \[2, 5, 1, 4, 3\]

   **Solution:**
   
   \[
   \begin{array}{c|c|c}
   \hline
   2 & 5, 1, 4 & 3 \\
   \hline
   2 & 5, 1 & 4, 3 \\
   \hline
   2, 1 & 5, 4 & 3 \\
   \hline
   2, 1 & 3, 4, 5 \\
   \hline
   \end{array}
   \]

2. **10 points** Write down an implementation of `max_heapify` in Python. The parameters are an array (Python list) `H` that represents a heap, an index `i`, and the number `n` of elements in the heap. The subtrees `left(i)` and `right(i)` are max heaps, but the node at index `i` may violate the max heap property. You may use the functions `left(i)`, `right(i)`, `parent(i)`, and `swap(H, i, j)` without defining them.

   **Solution:**
   
   ```python
   def max_heapify(H, i, n):
       l, r = left(i), right(i)
       if l < n and H[l] > H[i]:       # 4 points for finding largest
           largest = l
       else:
           largest = i
       if r < n and H[largest] < H[r]:
           largest = r
       if largest != i:
           swap(H, i, largest)          # 3 points
       max_heapify(H, largest)       # 3 points
   ```

3. **10 points** Given a hashtable with table size 4, insert the keys 6, 7, 8, 9, 10, 11 into the hashtable using the division method, \( h(k) = k \mod 4 \), and chaining to handle collisions.
Draw the resulting hashtable. How many collisions occurred during the insertion of the keys? What is the load factor of the resulting hashtable?

**Solution:**

0 \rightarrow 8
1 \rightarrow 9
2 \rightarrow 6 \rightarrow 10 \quad (6 \text{ points})
3 \rightarrow 7 \rightarrow 11

There were 2 collisions. 10 collided with 6 and 11 collided with 7. (2 points)
The load factor is 6/4 = 1.5. (2 points)