1. Consider the problem of sorting, in worst-case linear time, an array \( A \) of 10,000 9-digit social security numbers in increasing order. For each of the sorting algorithms below, indicate whether or not the algorithm will achieve worst-case, linear-time performance, and briefly explain why or why not.

(a) (2 points) CountingSort

(b) (2 points) RadixSort

(c) (2 points) BucketSort

(d) (2 points) QuickSort

2. (3 points) Explain how we can use the linear-time Select2 algorithm to implement a version of QuickSort whose worst-case running time is \( \Theta(n \lg n) \).
3. Consider a hash table of size $m = 12$ that uses collision-resolution by open addressing and the quadratic probing hash function $h(k, i) = ((k \mod m) + i + i^2) \mod m$.

(a) (4 points) Show the hash table resulting from inserting the keys 10, 22, 34 and 16, in this order.

(b) (4 points) The hash function for this hash table does not generate a valid probe sequence. Explain why and give a key that cannot be inserted into the table you produced in part (a) above.
4. Consider the following valid red-black tree, where “R” indicates a red node, and “B” indicates a black node. Note that the black dummy sentinel leaf nodes are not shown.

(a) (3 points) Show the resulting red-black tree after using RB-INSERT to insert key 3 into the above tree.

(b) (3 points) Show the resulting red-black tree after using RB-DELETE to delete key 15 from the original tree above.