1.(20%) Among the data structures below,

　　　heaps, hashmaps, hashtables, stacks, queues, deques, and binary search trees,

what is the most appropriate data structure for the following application? Give your explanations.

a. Counting word frequencies

b. Checking the matching of tags in HTML

c. Querying the set of keys which are smaller than a given key

d. Shortest job first CPU scheduling.

2.(20%) Answer the following questions:

a. What is a stable sorting algorithm?

b. Radix sort is the fastest sorting algorithm with $O(n)$ time complexity. Why is it not often used?

c. A hashtable is the fastest data structure with $O(1)$ access time. When is it not applicable?

d. Consider a stack based on a singly linked list with each node pointing to its next node. For this stack, why must the data enter and exit the list from the head instead of the tail?

3.(20%) Given a doubly linked list which has a `header` field for the dummy head node and a `trailer` field for the dummy tail node. Suppose that each node has a `next` field for the next node and a `prev` field for the previous node. Please complete the following task with proper pseudo code.

a. What are the conditions for testing if the list has exactly 1 and 2 data nodes respectively?

b. What are the steps in sequence for adding a node `new_node` to the head of the list?

(Hint: The condition for an empty list is `header.next == trailer` or `trailer.prev == header`.)

4.(20%) Given an array-based binary heap where each node has a key smaller than or equal to any children. Suppose that the following keys are added in order to this empty min-heap:

　　　21, 15, 36, 44, 10, 4, 9, 13, 29, 25.

a.　What is the content of the array for the min-heap?

b.　If a removal-of-minimum operation is performed, what is the new content of the array?

5.(20%) Insert the following sequence of keys in order to an initially empty binary search tree:

　　　7, 16, 25, 45, 5, 10, 18, 30, 50, 12, 1.

Suppose that each node of the tree contains a key greater than or equal to the left subtree, and smaller than or equal to the right subtree. Answer the following questions:

a.　What is the height of the binary search tree?

b.　What is the output key sequence when doing a postorder traversal of the binary search tree?

(Hint: The height of a tree is the number of nodes on the longest path from the root to a leaf.)