

Question 1 (20 Points) Provide a short answer to the following questions.

- (a) Briefly explain the pigeon hole principle.

- (b) What is the balance property all *Red-Black Trees* seek to maintain?

- (c) What is the benefit to using a binary tree over other data structures?

- (d) What is the benefit to using a hash table over other data structures?

- (e) When designing a hashing function, what are two properties the function should exhibit?

Question 2 (20 Points) Give an appropriate data structure for the following scenarios. Justify your answer.

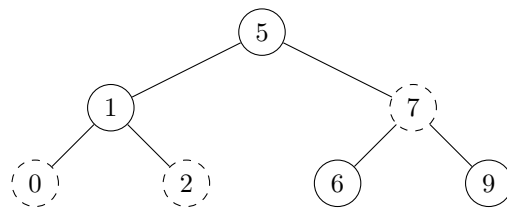
- (a) You have a large retail business for which you must keep track of inventory. Your inventory is outsourced to many different craftspeople who you allow to host items for sale on your website. You would like to record all items on sale site-wide such that they are easily searchable. Note: because people are setting up new shops (and removing old ones) daily, your inventory will change very frequently.

- (b) You are the newest software architect for *Vet Co.*, a veterinary multinational. You have been tasked with designing a system to record all records of pets currently being cared for. People are very loyal customers of *Vet Co.* so your records change infrequently.

Question 3 (20 Points) Build a proper *AVL Tree* given the following inputs. Show all steps. Include – at each step – the balance factor of each node.

4, 5, 6, 3, 2, 1

Question 4 (20 Points) Insert the value 3 into the following *Red-Black Tree*. Denote red nodes with a dashed outline, black nodes with a solid circle, and double-black nodes with a double-solid circle. Show all steps.



Question 5 (20 Points) For the following hash functions, state whether or not the function is “good”. Justify your answer.

- (a) Given any string of alpha-numeric characters, map all letters to integers in the range $[0, 26)$ such that $a = 0$, $b = 1$, and so on. Map all number characters to their corresponding integer. Take the index to be the sum of this set of integers. Assume this function is used in a hash table with collision chaining.

- (b) Given an integer identifier n , create an index $i = \lfloor \frac{n}{10} \rfloor$. Assume this function is used in a hash table with open addressing.