#1) Using a list it would be N/T since it has to look through all N inside any of the given T.

Using a BST is would be N/2T since it would only have to look through has of whatever is in any given T.

#2) A) I would much rather use a hash table since I can use the hash to only search a fraction of the entries by being able to check the index in which it should be in of the hash table.

B) I would think that the differences are just about negligible since they will both be allocating new space only as needed. The array itself would be the only difference but that shouldn't change too much.

C) I would probably use a hash table in most cases as it would do less work on average since it wouldn't have to search through as many things on any given search. The binary search tree only comes to mind very useful when I need something in order since the hash table does not maintain order.