**1. Palindrome**

**Steps:**

1. Take the string, length of string and pass it to a function

2. Check the first index and last index characters are equal

3. If true, remove that two characters and repeat step 1

4. If not go to step 6.

5. a. If length is one or zero return true go to step 7.

b. else go to step 6

6. Return false.

7. End.

**2. Anagram**

**Steps:**

1. Start.

2. Create a string.

3. Pass the string to a function along with first index (f) and last index (l) as parameters.

3. Check if f and l are equal

a. If true return array and go to step 9.

b. Else If not go to step 4.

4. Assign f to some k and repeat it until k is less than or equal to l.

5. Repeat from step 5.

6. Swap array[k] and array [f].

7. Fix first character and assign first index by 1 then go to step 1. (Recursive).

8. Go to step 6 (backtracking).

9. End.

**3. Binary search**

**Steps:**

1. Take array of elements

2. Pass an element to search and store it in a variable n

3. Declare and initialize left as first index, right as last index.

4. a. Check left is less than or equal to right if true go to step 5

b. Else go to step 7.

5. a. compute (left + right) and divide it by 2 and store it in middle

5 b. compare n with array [middle]

b. 1. If n is equal with middle of array then return middle

b. 2.Else go to step 6

6 A. If array [middle] is less than n

a. 1. Increment middle by 1 and assign it to left and go to step 4

6 B. else

b 1. Decrement middle by 1 and assign it to right and go to step 4

7. End.