1. How do you find the missing number in a given integer array of 1 to 100?
   Ans:

```
#include <stdio.h>

void Missing(int arr[], int N)

{

	int temp[N + 1];

	for (int i = 0; i <= N; i++) {

		temp[i] = 0;

	}


	for (int i = 0; i < N; i++) {

		temp[arr[i] - 1] = 1;

	}

	int ans;

	for (int i = 0; i <= N; i++) {

		if (temp[i] == 0)

			ans = i + 1;

	}

	printf("%d", ans);

}

int main()

{

	int arr[] = { 1, 3, 7, 5, 6, 2 };

	int n = sizeof(arr) / sizeof(arr[0]);

	Missing(arr, n);

}
```

2. How do you find the duplicate number on a given integer array?

```
		for (i = 0; i < size - 1; i++)
			for (j = i + 1; j < size; j++)
				if (arr[i] == arr[j])
					printf("%d ", arr[i]);
	}
```

```
int main()
{
	int arr[] = { 4, 2, 4, 5, 2, 3, 1 };
	int arr_size = sizeof(arr) / sizeof(arr[0]);
	duplicate(arr, arr_size);
	getchar();
	return 0;
}
```

3. How do you find the largest and smallest number in an unsorted integer array?

```
#include <stdio.h>
#include <stdlib.h>

int cmpfunc(const void* a, const void* b)

{
	return (*(int*)a - *(int*)b);
}

int kthSmallest(int arr[], int N, int K)
{
	// Sort the given array
	qsort(arr, N, sizeof(int), cmpfunc);

	// Return k'th element in the sorted array
	return arr[K - 1];
}

int main ()
{
	int arr[] = { 12, 3, 5, 7, 19 };
	int N = sizeof(arr) / sizeof(arr[0]), K = 2;

	// Function call
	printf("K'th smallest element is %d",
		kthSmallest(arr, N, K));
	return 0;
}
```

4.How do you find all pairs of an integer array whose sum is equal to a given number?

```
for (int i = 0; i < n; i++)

    for (int j = i + 1; j < n; j++)

        if (arr[i] + arr[j] == sum)

            System.out.println("(" + arr[i] + ", " + arr[j] + ")");
```

5. How do you find duplicate numbers in an array if it contains multiple duplicates?

```
for (int i = 0; i < len - 1; i++) {
    for (int j = i + 1; j < len; j++) {
        if (arr[i] == arr[j]) {
            if (al.contains(arr[i])) {
                break;
```

6. How are duplicates removed from a given array in Java?

```
    int[] temp = new int[n];
    int j = 0;
    for (int i=0; i<n-1; i++){
      if (arr[i] != arr[i+1]){
        temp[j++] = arr[i];
      }
     }
    temp[j++] = arr[n-1];
    for (int i=0; i<j; i++){
      arr[i] = temp[i];
    }
    return j;
  }

  public static void main (String[] args) {
    int arr[] = {10,20,20,30,30,40,50,50};
    int length = arr.length;
    length = removeDuplicateElements(arr, length);
    for (int i=0; i<length; i++)
      System.out.print(arr[i]+" ");
  }
}
```

7.Quick sort Algorithm?

```
static void swap(int[] arr, int i, int j)
{
        int temp = arr[i];
        arr[i] = arr[j];
        arr[j] = temp;
}
static int partition(int[] arr, int low, int high)
{
        int pivot = arr[high];
        int i = (low - 1);
        for (int j = low; j <= high - 1; j++) {
                if (arr[j] < pivot) {
                        i++;
                        swap(arr, i, j);
                }
        }
        swap(arr, i + 1, high);
        return (i + 1);
}
static void quickSort(int[] arr, int low, int high)
{
        if (low < high) {
                int pi = partition(arr, low, high);
                quickSort(arr, low, pi - 1);
                quickSort(arr, pi + 1, high);
        }
}
static void printArray(int[] arr, int size)
{
        for (int i = 0; i < size; i++)
```

8. . How do you remove duplicates from an array in place?

```
for (int i = 0; i < n-1; i++)
        if (arr[i] != arr[i+1])
           arr[j++] = arr[i];

     arr[j++] = arr[n-1];

     return j;
  }

  public static void main (String[] args)
  {
     int arr[] = {1, 2, 2, 3, 4, 4, 4, 5, 5};
     int n = arr.length;

     n = removeDuplicates(arr, n);
```

9. How do you reverse an array in place in Java?

```
System.out.println("Original Array:");
    for(int i=0;i<intArray.length;i++)
         System.out.print(intArray[i] + "  ");

    System.out.println();


    System.out.println("Original Array printed in reverse order:");
         for(int i=intArray.length-1;i>=0;i--)
         System.out.print(intArray[i] + "  ");
    }
}
```

10. How are duplicates removed from an array without using any library?

```
int j = 0;
     for (int i = 0; i < n - 1; i++)

       if (arr[i] != arr[i + 1])
          temp[j++] = arr[i];

     temp[j++] = arr[n - 1];


     for (int i = 0; i < j; i++)
       arr[i] = temp[i];

     return j;
```

11.. How do you print duplicate characters from a string?

```
int count[] = new int[NO_OF_CHARS];
      fillCharCounts(str, count);

      for (int i = 0; i < no_of_chars; i++)
         if (count[i] > 1)
            System.out.println((char)(i) +
                  ", count = " + count[i]);
```

12. How do you check if two strings are anagrams of each other?

```
int n1 = str1.length;
      int n2 = str2.length;

      if (n1 != n2)
         return false;

      Arrays.sort(str1);
      Arrays.sort(str2);

      for (int i = 0; i < n1; i++)
         if (str1[i] != str2[i])
            return false;

      return true;
   }
```

13. 23. How do you print the first non-repeated character from a string?

```
static void getCharCountArray(String str)
   {
      for (int i = 0; i < str.length(); i++)
         count[str.charAt(i)]++;
   }

     character in a string. If all characters are
     repeating then returns -1 */
   static int firstNonRepeating(String str)
   {
      getCharCountArray(str);
      int index = -1, i;

      for (i = 0; i < str.length(); i++) {
         if (count[str.charAt(i)] == 1) {
            index = i;
            break;
         }
      }

      return index;
```

14. How can a given string be reversed using recursion?

JAVA:

```
if ((str==null)||(str.length() <= 1))
        System.out.println(str);
      else
      {
         System.out.print(str.charAt(str.length()-1));
         reverse(str.substring(0,str.length()-1));
      }
   }

   public static void main(String[] args)
   {
      String str = "sadaiyappan";
      StringReverse obj = new StringReverse();
      obj.reverse(str);
```

C:

```
if (*str)
  {
     reverse(str+1);
     printf("%c", *str);
  }
}

int main()
{
  char a[] = "SADAIYAPPAN";
  reverse(a);
  return 0;
}
```

15. How do you check if a string contains only digits?

```
for (int i = 0; i < n; i++) {
          if (str.charAt(i) < '0'
             || str.charAt(i) > '9') {
             return false;
          }
      }
      return true;
   }

   public static void main (String args[])
   {
      String str = "1a234";
      int len = str.length();
```

16. How are duplicate characters found in a string?

```
for i in range(0, len(string)):
   count = 1;
   for j in range(i+1, len(string)):
```

```
        if(string[i] == string[j] and string[i] != ' '):
            count = count + 1;

            string = string[:j] + '0' + string[j+1:];
        if(count > 1 and string[i] != '0'):
        print(string[i]);
```

17. How do you count the number of vowels and consonants in a given string?

```
for (i = 0; str[i] != '\0'; i++) {

        ch = str[i];

        if (ch == 'a' || ch == 'e'
            || ch == 'i' || ch == 'o'
            || ch == 'u' || ch == 'A'
            || ch == 'E' || ch == 'I'
            || ch == 'O' || ch == 'U')
            vowels++;


        else if (ch == ' ')
            continue;

        else

            consonants++;
```

18. How do you count the occurrence of a given character in a string?

```
int count(string s, char c)
{

    int res = 0;

    for (int i=0;i<s.length();i++)


        if (s[i] == c)
            res++;

    return res;
}
```

19. How do you find all the permutations of a string?

```
for (i = l; i <= r; i++)
   {
      swap((a+l), (a+i));
      permute(a, l+1, r);
      swap((a+l), (a+i));
   }

int main()
{
   char str[] = "ABC";
   int n = strlen(str);
   permute(str, 0, n-1);
   return 0;
```

20. How do you reverse words in a given sentence without using any library method?

```
while (*temp) {
      temp++;
      if (*temp == '\0') {
         reverse(word_begin, temp - 1);
      }
      else if (*temp == ' ') {
         reverse(word_begin, temp - 1);
         word_begin = temp + 1;
```