CAT 1

NAME: JITHENDRA BOODATI

REG NO: 22BCE1947

1. **Write a C program to divide the integer array into two halves using function recursion. Call the user-defined "divide" function recursively, with the left half of the split array being passed as an argument for that function. Let the recursive function to get executed until the array size becomes one. Count the number of iterations to reach the base condition. Explain the working procedure of recursive function with stack structure**

CODE:

```
#include <stdio.h>

int count_iterations = 0;
void divide(int arr[], int start, int end) {
   if (start >= end) {
      return;
   }
   int mid = (start + end) / 2;
   divide(arr, start, mid);
   divide(arr, mid + 1, end);
   printf("Array from %d to %d: ", start, end);
   for (int i = start; i <= end; i++) {
      printf("%d ", arr[i]);
   }
   printf("\n");
   count_iterations++;
}

int main() {
```

```
    int arr[] = { 5, 3, 7, 1, 8, 2, 6, 4 };
    int size = sizeof(arr) / sizeof(arr[0]);
    divide(arr, 0, size- 1);
    printf("\nNumber of iterations: %d\n", count_iterations);
    return 0;
}
```

2. **Consider that you are going to analyze the characters in the given string. Write a C program to extract the characters in the given string and print whether the character is an uppercase alphabet, lowercase alphabet, digits, whitespace, special symbols. Print the count of each category by storing their counts in an array. Use appropriate looping constructs to implement this.**

CODE:

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

int main() {
    char str[] = "Hello World! 123";
    int count[5] = { 0 };
    for (int i = 0; str[i] != '\0'; i++) {
        char ch = str[i];
        if (isupper(ch)) {
            count[0]++;
            printf("%c is an uppercase alphabet.\n", ch);
        } else if (islower(ch)) {
            count[1]++;
            printf("%c is a lowercase alphabet.\n", ch);
        } else if (isdigit(ch)) {
            count[2]++;
            printf("%c is a digit.\n", ch);
        } else if (isspace(ch)) {
            count[3]++;
```

```
            printf("%c is a whitespace.\n", ch);
        } else {
            count[4]++;
            printf("%c is a special symbol.\n", ch);
        }
    }

    printf("\nCounts:\n");
    printf("Uppercase alphabets: %d\n", count[0]);
    printf("Lowercase alphabets: %d\n", count[1]);
    printf("Digits: %d\n", count[2]);
    printf("Whitespaces: %d\n", count[3]);
    printf("Special symbols: %d\n", count[4]);

    return 0;
}
```

3. **Write a c program to find the sum of the series 1!/1+2!/2+3!/3+4!/4+5!/5 ... n!/n by utilizing user defined recursive function? Get the value of n from the user. Do not use any storage classes. Without returning the calculated result from the function, display the result in main.**

CODE:

```
#include <stdio.h>

void factorial(int n, double *result) {
    if (n == 1) {
        *result = 1;
        return;
    }
    factorial(n-1, result);
    *result *= n;
```

```
int main() {
   int n;
   double sum = 0;
   printf("Enter the value of n: ");
   scanf("%d", &n);

   for (int i = 1; i <= n; i++) {
      double result;
      factorial(i, &result);
      sum += result/i;
   }
   printf("Sum of the series: %f\n", sum);
   return 0;
}
```

4. **ABC car showroom sells various types of cars such as Hatchback, Sedan, SUVs, and MUV. Due to the year-end sale, the showroom provides a 3%, 5%, 10%, and 15% discount for various car models Hatchback, Sedan, SUV, and MUV respectively. Also applies 12% of GST for the total amount of purchase Write a C program to implement the above scenario which will read the type_of_the_car, price_of_the_car and extra-fitting_price_of_the_car as input from the user and estimate the Net amount to be paid to the showroom.**
   If the type of car is other than Hatchback, Sedan, SUV, and MUV then display "Invalid Type". (Difficulty Level: Easy)
   The net amount to be paid to the showroom is estimated as follows: (For example-if the purchased car is Hatchback)
   Total = price_of_the_car + extra-fitting_price_of_the_car Discount = Total * 0.03 // 0.03 denotes 3% wastage gst = (Total - Discount) * 0.12 // 0.12 denotes 12% GST net = Total – Discount + gst

   CODE:

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

int main() {
```

```
    char type[10];
    float price, fitting, total, discount, gst, net;
    printf("Enter the type of the car (Hatchback/Sedan/SUV/MUV): ");
    scanf("%s", type);
    printf("Enter the price of the car: ");
    scanf("%f", &price);
    printf("Enter the extra fitting price of the car: ");
    scanf("%f", &fitting);

    total = price + fitting;
    if (strcmp(type, "Hatchback") == 0) {
        discount = total * 0.03;
    } else if (strcmp(type, "Sedan") == 0) {
        discount = total * 0.05;
    } else if (strcmp(type, "SUV") == 0) {
        discount = total * 0.10;
    } else if (strcmp(type, "MUV") == 0) {
        discount = total * 0.15;
    } else {
        printf("Invalid type\n");
        return 0;
    }
    gst = (total - discount) * 0.12;
    net = total - discount + gst;
    printf("Net amount to be paid: %0.2f\n", net);
    return 0;
}
```

5.Write a C Program that reads the input as a string from the user in main (). (1 mark)

The input should be a sentence with two words. Pass this string to a function.(1 mark)

Inside the function do the following operations:

• For the first word, keep only the first character of the word to be in upper case and the rest of the characters in lower case. (1 mark)
• For the second word, convert all the characters into upper case letter. (1 mark)
• Print the revised string consisting of the two words in the function itself (1 mark)
• Find the length of the entire string. Print its length in the function itself in the next line of the revised string. Use appropriate string function to print this result in the next line. (1 mark) • Return the length of the string, if the length is less than 20. Else return the size of the string. (2 marks)
Consider the input string given by the user is: • "computer programming"
• Revised string to be printed in the function is "Computer PROGRAMMING" What is the significant different between length and size of the string? What is the value that will get returned for the input string that is considered?

CODE:

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

int processString(char sentence[]) {
   int length = strlen(sentence);
   char firstWord[20], secondWord[20];
   scanf(sentence, "%s %s", firstWord, secondWord);
   firstWord[0] = toupper(firstWord[0]);
   for(int i = 1; i < strlen(firstWord); i++) {
      firstWord[i] = tolower(firstWord[i]);
   }
   for(int i = 0; i < strlen(secondWord); i++) {
      secondWord[i] = toupper(secondWord[i]);
   }

   printf("%s %s\n", firstWord, secondWord);
```

```
    printf("%d\n", length);
    else return length
    if(length >= 20) {
        return sizeof(sentence);
    } else {
        return length;
    }
}
int main() {
    char sentence[50];
    printf("Enter a sentence with two words: ");
    fgets(sentence, 50, stdin);
    int result = processString(sentence);
    printf("Result: %d\n", result);

    return 0;
}
```