>>>>> PAT 1

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

int isPrime(int num);
int isArmstrong(int num);
int isPerfect(int num);

int main() {
    int num;
    printf("Enter a positive integer: ");
    scanf("%d", &num);

    if (isPrime(num)) {
        printf("%d is a prime number\n", num);
    } else {
        printf("%d is not a prime number\n", num);
    }

    if (isArmstrong(num)) {
        printf("%d is an Armstrong number\n", num);
    } else {
        printf("%d is not an Armstrong number\n", num);
    }

    if (isPerfect(num)) {
        printf("%d is a perfect number\n", num);
    } else {
        printf("%d is not a perfect number\n", num);
    }

    return 0;
}

int isPrime(int num) {
    int i;
    for (i = 2; i <= sqrt(num); i++) {
        if (num % i == 0) {
            return 0;
        }
    }
    return 1;
}

int isArmstrong(int num) {
    int sum = 0;
    int temp = num;
    int digitCount = 0;

    while (temp != 0) {
        digitCount++;
        temp /= 10;
    }

    temp = num;
    while (temp != 0) {
        int remainder = temp % 10;
        sum += pow(remainder, digitCount);
        temp /= 10;
```

```
    }

    if (sum == num) {
        return 1;
    } else {
        return 0;
    }
}

int isPerfect(int num) {
    int i, sum = 0;
    for (i = 1; i <= num / 2; i++) {
        if (num % i == 0) {
            sum += i;
        }
    }
    if (sum == num) {
        return 1;
    } else {
        return 0;
    }
}
```

>>>>> PAT 2

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

int main() {
    char str[100];
    int vowels = 0, consonants = 0, spaces = 0, specialChars = 0, words =
1;

    printf("Enter a string: ");
    fgets(str, sizeof(str), stdin);

    for (int i = 0; str[i] != '\0'; i++) {
        if (isspace(str[i])) {
            spaces++;
            words++;
        } else if (isalpha(str[i])) {
            if (str[i] == 'a' || str[i] == 'e' || str[i] == 'i' || str[i]
== 'o' || str[i] == 'u' ||
                str[i] == 'A' || str[i] == 'E' || str[i] == 'I' || str[i]
== 'O' || str[i] == 'U') {
                vowels++;
            } else {
                consonants++;
            }
        } else {
            specialChars++;
        }
    }

    printf("Words = %d\n", words);
    printf("Vowels = %d\n", vowels);
```

```
    printf("Consonants = %d\n", consonants);
    printf("Space = %d\n", spaces);
    printf("Special Characters = %d\n", specialChars);

    return 0;
}
```

>>>>> CAT 1A

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

int main() {
    char str[100];
    int len, freq[256] = {0}, i;
    char *first_rep = NULL, *first_nonrep = NULL;

    printf("Enter a string: ");
    fgets(str, 100, stdin);
    len = strlen(str) - 1;

    for(i = 0; i < len; i++) {
        freq[(int)str[i]]++;
    }

    printf("Length of the string is: %d\n", len);
    printf("Word frequency is: ");
    for(i = 0; i < 256; i++) {
        if(freq[i] != 0) {
            printf("%c:%d ", i, freq[i]);
        }
    }
    printf("\n");

    for(i = 0; i < len; i++) {
        if(freq[(int)str[i]] > 1 && first_rep == NULL) {
            first_rep = &str[i];
        } else if(freq[(int)str[i]] == 1 && first_nonrep == NULL) {
            first_nonrep = &str[i];
        }

        if(first_rep != NULL && first_nonrep != NULL) {
            break;
        }
    }

    if(first_rep == NULL) {
        printf("No repeated characters found in the string.\n");
    } else {
        printf("First repeated character is: %c\n", *first_rep);
    }

    if(first_nonrep == NULL) {
        printf("No non-repeated characters found in the string.\n");
    } else {
        printf("First non-repeated character is: %c\n", *first_nonrep);
    }
```

```
    return 0;
}
```

>>>>> CAT 1B

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

typedef struct {
    char name[50];
    int age;
    char position[50];
    char date_of_joining[11];
} Employee;


int cmp_name(const void* a, const void* b) {
    return strcmp(((Employee*)a)->name, ((Employee*)b)->name);
}


int cmp_date(const void* a, const void* b) {
    return strcmp(((Employee*)a)->date_of_joining, ((Employee*)b)-
>date_of_joining);
}

int main() {
    int num_employees;


    printf("Enter the number of employees: ");
    scanf("%d", &num_employees);


    Employee* employees = (Employee*) malloc(num_employees *
sizeof(Employee));


    for (int i = 0; i < num_employees; i++) {
        printf("Enter details of employee %d:\n", i+1);
        printf("Name: ");
        scanf("%s", employees[i].name);
        printf("Age: ");
        scanf("%d", &employees[i].age);
        printf("Position: ");
        scanf("%s", employees[i].position);
        printf("Date of joining (dd/mm/yyyy): ");
        scanf("%s", employees[i].date_of_joining);
    }


    qsort(employees, num_employees, sizeof(Employee), cmp_name);


    printf("\nEmployee List sorted by name:\n");
```

```
    for (int i = 0; i < num_employees; i++) {
        printf("Name: %s\n", employees[i].name);
        printf("Age: %d\n", employees[i].age);
        printf("Position: %s\n", employees[i].position);
        printf("Date of Joining: %s\n\n", employees[i].date_of_joining);
    }


    qsort(employees, num_employees, sizeof(Employee), cmp_date);


    printf("\nEmployee List sorted by date of joining:\n");
    for (int i = 0; i < num_employees; i++) {
        printf("Name: %s\n", employees[i].name);
        printf("Age: %d\n", employees[i].age);
        printf("Position: %s\n", employees[i].position);
        printf("Date of Joining: %s\n\n", employees[i].date_of_joining);
    }

    free(employees);

    return 0;
}
```

G2 SLOT:::

>>>>>G1:::

```
#include <stdio.h>

int main() {
    float chennai[7], delhi[7], haveri[7], gangtok[7];
    float chennai_temp, delhi_temp, haveri_temp;
    int i;

    printf("Enter the temperature of Chennai for a week of 7 days:\n");
    for (i = 0; i < 7; i++) {
        scanf("%f", &chennai[i]);
    }

    for (i = 0; i < 7; i++) {
        delhi[i] = chennai[i] - 8;
        haveri[i] = chennai[i] + 5;
    }

    for (i = 0; i < 7; i++) {
        gangtok[i] = haveri[i] - delhi[i];
    }

    printf("Temperature of Gangtok for a week of 7 days:\n");
    for (i = 0; i < 7; i++) {
        printf("%.2f ", gangtok[i]);
    }
    printf("\n");

    return 0;
}
```

>>>>>G2

```
#include <stdio.h>

int sumOfDigits(int n) {
    int sum = 0;
    while (n > 0) {
        sum += n % 10;
        n /= 10;
    }
    return sum;
}

int main() {
    int sum = 0;
    for (int i = 1000; i <= 9998; i += 2) {
        sum += i;
    }
    while (sum > 9) {
        sum = sumOfDigits(sum);
    }
    if (sum % 2 == 0) {
        printf("Even found\n");
    } else {
        printf("Odd found\n");
    }
    return 0;
}
```

>>>>> G3

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

int main()
{
    char password[] = "aeiceg";
    char input[3][3];
    char diagonal[5];
    int i, j, k = 0;

    printf("Enter the input characters:\n");
    for(i=0; i<3; i++)
    {
        for(j=0; j<3; j++)
        {
            scanf(" %c", &input[i][j]);
        }
    }

    diagonal[k++] = input[0][0];
    diagonal[k++] = input[1][1];
    diagonal[k++] = input[2][2];
    diagonal[k++] = input[0][2];
    diagonal[k] = input[2][0];

    if(strcmp(diagonal, password) == 0)
    {
        printf("Password matched. The locker is opened successfully.\n");
```

```
    }
    else
    {
        printf("Password not matched. The locker cannot be opened.\n");
    }

    return 0;
}
```

>>>>>G4

```
#include <stdio.h>

int main() {
    int original_marks[25], revised_marks[25], birth_month[25];
    float original_total = 0, revised_total = 0, original_avg,
revised_avg;

        printf("Enter original marks and birth month for each
student:\n");
    for(int i=0; i<25; i++) {
        printf("Student %d: ", i+1);
        scanf("%d %d", &original_marks[i], &birth_month[i]);
    }

    for(int i=0; i<25; i++) {
        revised_marks[i] = original_marks[i] + birth_month[i];
        original_total += original_marks[i];
        revised_total += revised_marks[i];
    }

    original_avg = original_total / 25;
    revised_avg = revised_total / 25;

    printf("Class average for original marks: %.2f\n", original_avg);
    printf("Class average for revised marks: %.2f\n", revised_avg);

    if(revised_avg - original_avg >= 5) {
        printf("Can implement - Significant increase in class
average.\n");
    } else {
        printf("Need not implement - No significant increase in class
average.\n");
    }

    return 0;
}
```

>>>>>G5

```
#include <stdio.h>

int power(int x, int n);

int main() {
    int x, n, result;
```

```
    printf("Enter a number x: ");
    scanf("%d", &x);

    printf("Enter a number n (less than or equal to 5): ");
    scanf("%d", &n);

    result = power(x, n);

    printf("%d to the power of %d is %d\n", x, n, result);

    return 0;
}

int power(int x, int n) {
    if (n == 0) {
        return 1;
    } else {
        return x * power(x, n-1);
    }
}
```

>>>>>>>>cC2 SLOT

>>>>>C1

```
#include <stdio.h>

void divide(int arr[], int left, int right, int* count);

int main() {
    int arr[] = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10};
    int count = 0;
    divide(arr, 0, 9, &count);
    printf("Number of iterations: %d\n", count);
    return 0;
}

void divide(int arr[], int left, int right, int* count) {
    if (left == right) {
        return;
    }
    int mid = (left + right) / 2;
    divide(arr, left, mid, count);
    divide(arr, mid+1, right, count);
    (*count)++;
    printf("Iteration %d: left=%d, right=%d\n", *count, left, right);
}
```

>>>>>C2

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

int main() {
    char str[100];
    int counts[5] = {0};
```

```
    printf("Enter a string: ");
    fgets(str, 100, stdin);

    for (int i = 0; str[i] != '\0'; i++) {
        char c = str[i];
        if (isupper(c)) {
            counts[0]++;
            printf("%c is an uppercase alphabet.\n", c);
        }
        else if (islower(c)) {
            counts[1]++;
            printf("%c is a lowercase alphabet.\n", c);
        }
        else if (isdigit(c)) {
            counts[2]++;
            printf("%c is a digit.\n", c);
        }
        else if (isspace(c)) {
            counts[3]++;
            printf("%c is a whitespace.\n", c);
        }
        else {
            counts[4]++;
            printf("%c is a special symbol.\n", c);
        }
    }

    printf("Counts:\n");
    printf("Uppercase alphabet: %d\n", counts[0]);
    printf("Lowercase alphabet: %d\n", counts[1]);
    printf("Digit: %d\n", counts[2]);
    printf("Whitespace: %d\n", counts[3]);
    printf("Special symbol: %d\n", counts[4]);

    return 0;
}


>>>>>>C3

#include <stdio.h>

void findFactorial(int n, double* result) {
    double factorial = 1;
    for (int i = 1; i <= n; i++) {
        factorial *= i;
    }
    *result += factorial/n;
    if (n > 1) {
        findFactorial(n - 1, result);
    }
}

int main() {
    int n;
    double sum = 0;

    printf("Enter the value of n: ");
    scanf("%d", &n);
```

```
    findFactorial(n, &sum);

    printf("The sum of the series is: %.2lf", sum);

    return 0;
}
```

>>>>>C4

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

int main() {
    char type[10];
    float price, extra, discount, gst, net, total;

    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", &extra);

    total = price + extra;

    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 * 0.1;
    }
    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: %.2f\n", net);

    return 0;
}
```

>>>>>C5

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

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

int process_string(char* str);

int main() {
    char str[100];
    printf("Enter a sentence with two words: ");
    scanf("%[^\n]", str);
    int len = process_string(str);
    printf("Length of string: %d\n", len);
    return 0;
}

int process_string(char* str) {
    char first[50], second[50];
    int i = 0, j = 0, k = 0;
    while (str[i] != ' ') {
        first[j++] = toupper(str[i++]);
    }
    first[j] = '\0';
    i++;
    while (str[i] != '\0') {
        second[k++] = toupper(str[i++]);
    }
    second[k] = '\0';
    printf("%s %s\n", first, second);
    int len = strlen(str);
    printf("Length of revised string: %d\n", len);
    if (len < 20) {
        return len;
    }
    else {
        return sizeof(str);
    }
}

<<15 question>>

#include <stdio.h>
int main() {
int i, j, a, b, c, sum, acute = 0, right = 0, obtuse = 0, wrong = 0;
    for (i = 1; i <= 5; i++) {
        printf("Enter the three angles of triangle %d:\n",i);
        scanf("%d %d %d", &a, &b, &c);
        sum = a + b + c;
        if (sum > 180) {
            printf("Wrong Entry try again\n");
            wrong++;
            i--;
            continue;
        }
        if (a < b) {
            j = a;
    a = b;
            b = j;
        }
        if (a < c) {
            j = a;
            a = c;
```

```
            c = j;
        }
        if (a*a == b*b + c*c) {
            printf("Right-Angled Triangle\n");
            right++;
        } else if (a*a < b*b + c*c) {
            printf("Acute Angled Triangle\n");
            acute++;
        } else {
            printf("Obtuse Angled Triangle\n");
            obtuse++;
        }
    }
    printf("\nAcute Angled Triangle: %d\n", acute);
    printf("Right Angled Triangle: %d\n", right);
    printf("Obtuse Angled Triangle: %d\n", obtuse);
    printf("Wrong Entries: %d\n", wrong);
Â Â Â Â returnÂ 0;
}
```