

Problems in C Programming(2347207)

1. Write a C program to read the value of an integer m and display the value of n is 1 when m is larger than 0, 0 when m is 0 and -1 when m is less than 0.

```
#include <stdio.h>

int main() {

    int m, n;

    // Read the value of integer m

    printf("Enter the value of m: ");

    scanf("%d", &m);

    // Determine the value of n based on the value of m

    if (m > 0) {

        n = 1;

    } else if (m == 0) {

        n = 0;

    } else {

        n = -1;

    }

    // Display the value of n

    printf("The value of n is: %d\n", n);

    return 0;

}
```

Output

```
/tmp/H0YMqpAf8S.o  
Enter the value of m: 5  
The value of n is: 1
```

Output

```
/tmp/H0YMqpAf8S.o  
Enter the value of m: -12  
The value of n is: -1
```

2. Write a C program to read the age of a candidate and determine whether he is eligible to cast his/her own vote.

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

```
int main() {  
    int age;
```

```
    printf("Enter your age: ");  
    scanf("%d", &age);
```

```
    if (age >= 18) {  
        printf("Congratulations! You are eligible to cast your vote.\n");  
    } else {  
        printf("Sorry, you are not eligible to cast your vote.\n");  
        printf("You must be at least 18 years old to vote.\n");  
    }  
}
```

```
    return 0;  
}
```

Output

```
/tmp/H0YMqpAf8S.o  
Enter your age: 21  
Congratulations! You are eligible to cast your vote.
```

3. Write a C program to read the roll no, name and Semester Marks for 5 Subjects(each Semester) for 3 students and calculate the total percentage. The conditions must be followed

- a. If Percentage ≤ 40 then and Check all the subjects marks if marks ≤ 40 then Print Result Fail
- b. Percentage ≥ 60 and ≤ 80 then Print Result as First Class
- c. Percentage ≥ 80 then ≤ 90 Print Result as First Class with Distinction
- d. Percentage $\geq 90 - 100$ then Print Result as Outstanding
- e. Print the Cumulative Mark sheet for all 6 Semesters with each semester 5 subjects and check eligibility to get the degree

```
#include <stdio.h>
struct Student {
    int rollNo;
    char name[50];
    int marks[5];
};
float calculatePercentage(int marks[], int numSubjects) {
    int totalMarks = 0;
    for (int i = 0; i < numSubjects; i++) {
        totalMarks += marks[i];
    }
    return (float)totalMarks / (numSubjects * 100) * 100;
}
void printResult(float percentage) {
    if (percentage <= 40) {
        printf("Result: Fail\n");
    } else if (percentage >= 60 && percentage <= 80) {
        printf("Result: First Class\n");
    } else if (percentage > 80 && percentage <= 90) {
        printf("Result: First Class with Distinction\n");
    } else if (percentage > 90 && percentage <= 100) {
        printf("Result: Outstanding\n");
    }
}
int main() {
    struct Student students[3];
    int numSubjects = 5;
    int numStudents = 3;
    for (int i = 0; i < numStudents; i++) {
        printf("Enter details for student %d:\n", i + 1);
        printf("Roll No: ");
        scanf("%d", &students[i].rollNo);
        printf("Name: ");
```

```

scanf("%s", students[i].name);
for (int j = 0; j < numSubjects; j++) {
    printf("Enter marks for subject %d: ", j + 1);
    scanf("%d", &students[i].marks[j]);
}
printf("\n");
}

printf("Cumulative Mark Sheet:\n");
for (int i = 0; i < numStudents; i++) {
    float percentage = calculatePercentage(students[i].marks,
numSubjects);
    printf("Student %d (Roll No: %d, Name: %s)\n", i + 1, students[i].rollNo,
students[i].name);
    printf("Percentage: %.2f%%\n", percentage);
    printResult(percentage);
    printf("\n");
}
return 0;
}

```

Output

```

/tmp/HOYMqpAf8S.o
Enter details for student 1:
Roll No: 7
Name: Alwin
Enter marks for subject 1: 80
Enter marks for subject 2: 77
Enter marks for subject 3: 92
Enter marks for subject 4: 83
Enter marks for subject 5: 75
Enter details for student 2:
Roll No: 28
Name: Spoorthi
Enter marks for subject 1: 90
Enter marks for subject 2: 92
Enter marks for subject 3: 88
Enter marks for subject 4: 97
Enter marks for subject 5: 83
Enter details for student 3:
Roll No: 11
Name: Jeevan
Enter marks for subject 1: 85
Enter marks for subject 2: 88
Enter marks for subject 3: 73
Enter marks for subject 4: 95
Enter marks for subject 5: 74

```

```

Cumulative Mark Sheet:
Student 1 (Roll No: 7, Name: Alwin)
Percentage: 81.40%
Result: First Class with Distinction

Student 2 (Roll No: 28, Name: Spoorthi)
Percentage: 90.00%
Result: First Class with Distinction

Student 3 (Roll No: 11, Name: Jeevan)
Percentage: 83.00%
Result: First Class with Distinction

```

4. Write a program in C to create simple Calculator using Menu Driven (Switch Case)
#include <stdio.h>

```
int main() {
    int choice;
    float num1, num2, result;

    printf("Menu Driven Calculator\n");
    printf("1. Addition\n");
    printf("2. Subtraction\n");
    printf("3. Multiplication\n");
    printf("4. Division\n");
    printf("5. Exit\n");

    while (1) {
        printf("\nEnter your choice (1-5): ");
        scanf("%d", &choice);

        if (choice == 5) {
            printf("Exiting the calculator.\n");
            break;
        }

        printf("Enter two numbers: ");
        scanf("%f %f", &num1, &num2);

        switch (choice) {
            case 1:
                result = num1 + num2;
                printf("Addition: %.2f\n", result);
                break;
            case 2:
                result = num1 - num2;
                printf("Subtraction: %.2f\n", result);
                break;
            case 3:
                result = num1 * num2;
                printf("Multiplication: %.2f\n", result);
                break;
            case 4:
                if (num2 != 0) {
                    result = num1 / num2;
                    printf("Division: %.2f\n", result);
                }
            default:
                printf("Invalid choice\n");
        }
    }
}
```

```

        } else {
            printf("Error: Cannot divide by zero.\n");
        }
        break;
    default:
        printf("Invalid choice! Please choose a valid option.\n");
    }
}

return 0;
}

```

Output

```
/tmp/H0YMqpAf8S.o
```

```
Menu Driven Calculator
```

```
1. Addition
```

```
2. Subtraction
```

```
3. Multiplication
```

```
4. Division
```

```
5. Exit
```

```
Enter your choice (1-5): 1
```

```
Enter two numbers: 5 2
```

```
Addition: 7.00
```

```
Enter your choice (1-5): 2
```

```
Enter two numbers: 5 2
```

```
Subtraction: 3.00
```

```
Enter your choice (1-5): 3
```

```
Enter two numbers: 5 2
```

```
Multiplication: 10.00
```

```
Enter your choice (1-5): 4
```

```
Enter two numbers: 5 2
```

```
Division: 2.50
```

```
Enter your choice (1-5): 5
```

```
Exiting the calculator.
```

5. Write a program in C to make a pyramid pattern with numbers increased by multiples of five

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

```
int main() {  
    int i, space, rows, k = 0;  
  
    printf("Enter the number of rows: ");  
  
    scanf("%d", &rows);  
  
    for (i = 1; i <= rows; ++i, k = 0) {  
        for (space = 1; space <= rows - i; ++space) {  
            printf(" ");  
        }  
        while (k != 2 * i - 1) {  
            printf("%-3d", 5 * k);  
            ++k;  
        }  
        printf("\n");  
    }  
    return 0;  
}
```

Output

```
/tmp/18qvUvmw64.o  
Enter the number of rows: 4  
0  
    0  5  10  
    0  5  10 15 20  
0  5  10 15 20 25 30
```

6. C program to calculate the tax of n given number of employees. Use a separate function to calculate the tax. Tax is 20% of basic if basic is less than 9000 otherwise tax is 25% of basic. dd Deductions of Savings around 2,00,000.

```
#include <stdio.h>

float calculateTax(float basic) {

    float deductions = 200000; // Deductions for savings up to 2,00,000
    if (basic < 9000) {
        return basic * 0.20 - deductions;
    } else {
        return basic * 0.25 - deductions;
    }
}

int main() {
    int n;
    printf("Enter the number of employees: ");
    scanf("%d", &n);
    float basic, tax;
    for (int i = 1; i <= n; i++) {
        printf("\nEnter basic salary for employee %d: ", i);
        scanf("%f", &basic);

        // Call the function to calculate tax
        tax = calculateTax(basic);

        printf("Tax for employee %d: %.2f\n", i, tax);
    }

    return 0;
}
```

Output

```
/tmp/18qvUvmw64.o
Enter the number of employees: 2
Enter basic salary for employee 1: 20
00000Tax for employee 1: 300000.00

Enter basic salary for employee 2: 3000000
Tax for employee 2: 550000.00
```


7. C program to calculate the salary Slip of a given number of employees.(DA, Basic Pay, HRA, TA, LOP)

```
#include <stdio.h>

void calculateSalarySlip(float basic, float* da, float* hra, float* ta, float* lop) {

    // Constants for DA, HRA, and TA percentages

    const float DA_PERCENTAGE = 0.10;

    const float HRA_PERCENTAGE = 0.15;

    const float TA_PERCENTAGE = 0.05;


    // Calculating DA, HRA, and TA based on the basic salary

    *da = basic * DA_PERCENTAGE;

    *hra = basic * HRA_PERCENTAGE;

    *ta = basic * TA_PERCENTAGE;


    // Assuming LOP as 1000 for demo purposes

    *lop = 1000;

}


int main() {

    int n;

    printf("Enter the number of employees: ");

    scanf("%d", &n);


    float basic, da, hra, ta, lop, totalSalary;
```

```
for (int i = 1; i <= n; i++) {  
    printf("\nEnter basic salary for employee %d: ", i);  
    scanf("%f", &basic);  
  
    // Call the function to calculate salary components  
    calculateSalarySlip(basic, &da, &hra, &ta, &lop);  
  
    // Calculate total salary  
    totalSalary = basic + da + hra + ta - lop;  
  
    // Print the salary slip for the employee  
    printf("\n--- Salary Slip for Employee %d ---\n", i);  
    printf("Basic Salary: %.2f\n", basic);  
    printf("DA: %.2f\n", da);  
    printf("HRA: %.2f\n", hra);  
    printf("TA: %.2f\n", ta);  
    printf("LOP: %.2f\n", lop);  
    printf("Total Salary: %.2f\n", totalSalary);  
}  
  
return 0;  
}
```

Output

```
/tmp/18qvUvmw64.o
Enter the number of employees: 2
Enter basic salary for employee 1: 200000
--- Salary Slip for Employee 1 ---
Basic Salary: 200000.00
DA: 20000.00
HRA: 30000.00
TA: 10000.00
LOP: 1000.00
Total Salary: 259000.00

Enter basic salary for employee 2: 24000
--- Salary Slip for Employee 2 ---
Basic Salary: 24000.00
DA: 2400.00
HRA: 3600.00
TA: 1200.00
LOP: 1000.00
Total Salary: 30200.00
```

8. Write a C program to print all the Armstrong numbers between two intervals.

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

// Function to calculate the number of digits in a number
int countDigits(int num) {
    int count = 0;
    while (num != 0) {
        num /= 10;
        count++;
    }
    return count;
}

// Function to check if a number is an Armstrong number
int isArmstrong(int num) {
    int originalNum = num;
    int numDigits = countDigits(num);
    int sum = 0;
```

```

while (num != 0) {
    int digit = num % 10;
    sum += pow(digit, numDigits);
    num /= 10;
}

return (sum == originalNum);
}

int main() {
    int start, end;

    printf("Enter the starting interval: ");
    scanf("%d", &start);

    printf("Enter the ending interval: ");
    scanf("%d", &end);

    printf("Armstrong numbers between %d and %d are:\n", start, end);

    for (int i = start; i <= end; i++) {
        if (isArmstrong(i)) {
            printf("%d\n", i);
        }
    }

    return 0;
}

```

Output

```

/tmp/18qvUvmw64.o
Enter the starting interval: 50
Enter the ending interval: 200
Armstrong numbers between 50 and 200 are:
153

```

9. Write a C program to find the 'Perfect' numbers within a 10 to 100

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

```
// Function to check if a number is a perfect number
```

```
int isPerfect(int num) {
```

```
    int sum = 1; // Start with 1 as all numbers are divisible by 1
```

```
    for (int i = 2; i * i <= num; i++) {
```

```
        if (num % i == 0) {
```

```
            sum += i;
```

```
            if (num / i != i) {
```

```
                sum += num / i;
```

```
            }
```

```
        }
```

```
    }
```

```
    return (sum == num);
```

```
}
```

```
int main() {
```

```
    int start = 10, end = 100;
```

```
    printf("Perfect numbers between %d and %d are:\n", start, end);
```

```
    for (int i = start; i <= end; i++) {
```

```

        if (isPerfect(i)) {
            printf("%d\n", i);
        }
    }

    return 0;
}

```

Output

```

/tmp/0BJkaItSP.o
Perfect numbers between 10 and 100 are:
28

```

10. Write a C program to convert given ASCII Charcter into number

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

```

int main() {
    char asciiChar;
    int asciiValue;

    printf("Enter an ASCII character: ");
    scanf("%c", &asciiChar);

    asciiValue = (int)asciiChar;

    printf("The ASCII value of '%c' is: %d\n", asciiChar, asciiValue);
}

```

```
    return 0;
}
```

Output

```
/tmp/18qvUvmw64.o
Enter an ASCII character: F
The ASCII value of 'F' is: 70
```

11. Write a C Program to use only an addition, how do you add eight 8's and get the number 1000?

```
#include <stdio.h>

int main() {
    int result = 888 + 88 + 8 + 8 + 8;
    printf("Result: %d\n", result);
    return 0;
}
```

Output

```
/tmp/18qvUvmw64.o
Result: 1000
```

12 Write a C Program to use only For loop to solve addition of n even numbers
#include <stdio.h>

```
int main() {
    int n, sum = 0;

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

```
// Loop through n even numbers and add them to the sum
for (int i = 2; i <= 2 * n; i += 2) {
    sum += i;
}

printf("Sum of first %d even numbers is: %d\n", n, sum);

return 0;
}
```

Output

```
/tmp/3E2MN6MoQa.o
Enter the value of n: 10
Sum of the first 10 even numbers: 110
```