
Q1) Write a C program to print numbers from 0 to 100. (You are required to write 3 separate answers each using While, Do..While, For, looping structures).

//While loop

```
#include <stdio.h>

int main()
{
    int x=0;
    while (x <= 100)
    {
        printf("%d \n",x);
        x++;
    }
}
```

//Do While loop

```
#include <stdio.h>

int main()
{
    int x = 0;
    do
    {
        printf("%d \n", x);
        x++;
    }
    while (x<=100);

}
```

//For loop

```
#include <stdio.h>

int main ()
{
    int x;
    for(x=0; x<=100; x++ )
        printf("%d \n", x);
}
```

Q2) Write a C program to calculate and print the total of 10 marks and the average. If the average is less than 50 program should print “Fail!” otherwise “Pass!”

```
#include <stdio.h>

int main()
{
    int mark=0, total=0, avg, counter;
    for (counter = 1; counter <= 10; counter++)
    {
        printf("Enter the %d Paper Marks :", counter);
        scanf("%d", &mark);
        total += mark; //...Total of marks
    }
    avg = total/10;
    printf("Average of Marks is %d \n", avg);
    //...If condition
    if(avg>=50)
    {
        printf("PASS");
    }
    else
    {
        printf("FAIL");
    }
}
```

}

}

Q3) Write a C program to calculate factorial of a user given number.

Hint:

- ☐ Select an appropriate looping structure.
- ☐ Factorial of '0' is '1' ($0! = 1$)
- ☐ Ex: factorial of number 5 is calculated as $5! = 5*4*3*2*1$

```
#include <stdio.h>

int main()
{
    int x,no,fac=1;

    printf("Enter Number:");
    scanf("%d",&no);

    if(no < 0)
    {
        printf("Error");
    }
    else{
        for(x=1;x<=no;x++)
            fac*=x;
    }
    printf("factorial = %d",fac);

}
```

Q4) Write a C program to calculate the sum of all digits of a user given number.

- ☐ If user input 123 your program should output 6. (calculated as $1+2+3$)

```

#include <stdio.h>

int main()
{
    int num, digit, sum = 0;

    printf("Enter the numbers: ");
    scanf("%d", &num);

    while (num > 0)
    {
        digit = num % 10;
        sum += digit;
        num /= 10;
    }

    printf("Sum of the digits: %d\n", sum);
}

```

Q5) Write a C program to reverse the digits of a number using *do-while* statement.

```

#include <stdio.h>

int main()
{
    int num, reversedNum = 0;

    printf("Enter the numbers: ");
    scanf("%d", &num);

    do
    {

```

```

    int digit = num % 10;
    reversedNum = reversedNum * 10 + digit;
    num /= 10;
}
while (num != 0);

printf("Reversed number: %d\n", reversedNum);
}

```

Q6) Write a C program to calculate nth power of a given integer. The user input base and exponent. (Do NOT use inbuilt functions, instead use a loop)

```

#include <stdio.h>
int main()
{
    int base, exponent, result = 1;

    printf("Enter the base: ");
    scanf("%d", &base);

    printf("Enter the exponent: ");
    scanf("%d", &exponent);

    for (int i = 0; i < exponent; i++)
    {
        result *= base;
    }
    printf("%d raised to the power %d is: %d\n", base, exponent, result);
}

```

Q7) Write a C program to print first 10 numbers of “Fibonacci Sequence”.

```

#include <stdio.h>

int main()
{
    int n = 10;
    int first = 0, second = 1, next;
    printf("Fibonacci Sequence: ");

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

    {
        if (i <= 1)
            next = i;
        else
        {
            next = first + second;
            first = second;
            second = next;
        }
        printf("%d ", next);
    }
    printf("\n");
}

```

Q8) Write a C program to check whether a given number is an Armstrong Number! (Refer to previous flowcharts)

```

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

int isArmstrong(int number);

int main() {

```

```
int number;

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

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

return 0;
}

int isArmstrong(int number) {
    int originalNumber, remainder, result = 0, n = 0;

    originalNumber = number;

    // Count the number of digits
    while (originalNumber != 0) {
        originalNumber /= 10;
        ++n;
    }

    originalNumber = number;

    // Calculate the sum of nth power of individual digits
    while (originalNumber != 0) {
        remainder = originalNumber % 10;
        result += pow(remainder, n);
        originalNumber /= 10;
    }
```

```
// Check if the result is equal to the original number
if (result == number)
    return 1;
else
    return 0;
}
```

Q9) Write a C program to print all the ASCII values for letters A to Z.

```
#include <stdio.h>
int main()
{
    char letter;

    printf("ASCII values for letters A to Z:\n");
    for (letter = 'A'; letter <= 'Z'; letter++)
    {
        printf("%c: %d\n", letter, letter);
    }
}
```

Q10) Write a program to print this pattern.

```
*
**
***
****
*****
```

```
#include<stdio.h>
int main()
```



```
int i,j;
for (i = 1; i <= 5; i++)
{
    for (j = 1; j <= 5; j++)
    {
        printf("*");
    }
    printf("\n");
}
```

Q11) Write a program to check whether a given number is prime or not.

```
#include <stdio.h>

int main() {

    int no, i, flg = 0;
    printf("Enter a positive integer: ");
    scanf("%d", &no);

    if (no == 0 || no == 1)
        flg = 1;

    for (i = 2; i <= no / 2; ++i)
    {
        if (no % i == 0)
        {
            flg = 1;
            break;
        }
    }
}
```

```
    if (flg == 0)
        printf("%d is a prime number.", no);
    else
        printf("%d is not a prime number.", no);

}
```

Q12) Write a C program to print all factors of a given integer.

```
#include <stdio.h>

int main()
{
    int no, i;
    printf("Enter a positive integer: ");
    scanf("%d", &no);
    printf("Factors of %d are: ", no);
    for (i = 1; i <= no; ++i)
    {
        if (no % i == 0)
        {
            printf("%d ", i);
        }
    }
}
```

Q12) Write a C program to add all user inputs until user input '-1'. And then display the sum.

```
#include <stdio.h>

int main() {
    int no, i, sum = 0;

    printf("Enter a positive integer: ");
    scanf("%d", &no);

    for (i = -1; i <= no; ++i)
    {
        sum += i;
    }

    printf("Sum = %d", sum);
}
```

Q13) Write a C program to read user inputs for an integer array (size = 10) and print the array.

```
#include <stdio.h>

int main()
{
    int i,arr[10];
    //input
    for(i=0;i<10;i++)
    {
        printf("Enter a value to the element %d ",i+1);
        scanf("%d",&arr[i]);
    }
    //display
    for(i=0;i<10;i++)
        printf("%d",arr[i]);
}
```

Q14) Re-Write the above code to count all the even numbers in above integer array and display the count.

```
#include<stdio.h>

int main()
{
    int i, a[10], evenC = 0;

    //input
    for(i=0; i<10; i++)
    {
        printf("Enter a value to the element %d ", i+1);

        scanf("%d", &a[i]);
    }

    //display
    for(i = 0; i < 10; i++)
    {
        if(a[i] % 2 == 0) evenC++;
    }

    printf("\n Total Number of Even Numbers = %d ", evenC);
}
```

Section B

1. Input 10 numbers and to output number of positive, number of negative, number of zeros.

```

#include <stdio.h>
int main()
{
    int no,counter=1,p=0,n=0,z=0;
    while(counter<=10)
    {
        printf("Enter %d number ",counter);
        scanf("%d",&no);
        if(no>0)
            p++;
        else if(no<0)
            n++;
        else
            z++;
        counter++;
    }
    printf("Total number of positives %d \n",p);
    printf("Total number of negatives %d \n",n);
    printf("Total number of zeros %d \n",z);
}

```

2. Input Marks of 10 students and output the maximum , minimum and average Marks.

```

#include <stdio.h>
int main()
{
    float mrk[10],sum=0,avg,max=0,min=0;
    int i;

    //input
    for(i=0;i<10;i++)
    {
        printf("Enter %d student marks: ",i+1);
        scanf("%f", &mrk[i]);
    }
}

```

```

//display
for (i = 1; i <10; i++)
{
    if(mrk[i]>max)
        max=mrk[i];
    else
        min=mrk[i];

    sum = sum+mrk[i];
}
avg=sum/10.0;
printf("The maximum mark is %.2f\n", max);
printf("The minimum mark is %.2f\n", min);
printf("The average mark is %.2f\n", avg);
}

```

3. Input price of 10 items and display the average value of an Item , number of items which the price is greater than 200.

```

#include <stdio.h>

int main()
{
    float prices[10],sum = 0,count = 0,avg;

    printf("Enter the prices of 10 items:\n");

    // Input
    for (int i = 0; i < 10; i++)
    {
        printf("Item %d: Rs", i + 1);
    }
}

```

```

    scanf("%f", &prices[i]);
    sum += prices[i];
    if (prices[i] > 200) {
        count++;
    }
}

// Calculate average
avg = (float)sum / 10.0;

// Display
printf("Average value of an item: $%.2f\n", avg);
printf("Number of items with a price greater than 200: %d\n", count);
}

```

4. Input the Employee no and the Basic Salary of the Employees in an organisation ending with the dummy value -999 for Employee no and count the number Employees whose Basic Salary ≥ 5000 .

```

#include <stdio.h>

int main()
{
    int emp_no, count = 0;
    float bsal;

    while (1)
    {
        printf("Enter Employee No (-999 to exit): ");
        scanf("%d", &emp_no);

        if (emp_no == -999)break;
    }
}

```

```

printf("Enter Basic Salary: ");
scanf("%f", &bsal);

if (bsal >= 5000)
    count++;
}

printf("Number of employees with Basic Salary >= 5000: %d\n", count);
}

```

5. Input employee number, and hours worked by employees, and to display the following:

Employee number, Over Time Payment, and the percentage of employees whose Over Time Payment exceeding the Rs. 4000/-.

The user should input -999 as employee number to end the program, and the normal Over Time Rate is Rs.150 per hour and Rs. 200 per hour for hours in excess of 40.

```

#include <stdio.h>

#define overtime_rate 150 // Rs.150 per hour
#define overtime_rate_excess 200 // Rs.200 per hour for excess hours

int main()
{
    int emp_no, hours, tot_emp = 0, emp_exc_4000 = 0;

    while (1) {
        printf("Enter employee number (or -999 to end): ");
        scanf("%d", &emp_no);

        if (emp_no == -999)
        {
            break;

```



```

    }

    printf("Enter hours worked by the employee: ");
    scanf("%d", &hours);

    tot_emp++;

    // Calculate overtime payment
    int overtime_hours = (hours > 40) ? (hours - 40) : 0;

    int overtime_payment = (overtime_hours * overtime_rate) + ((overtime_hours - 40) *
    overtime_rate_excess);

    // Check if overtime payment exceeds Rs. 4000
    if (overtime_payment > 4000)
    {
        emp_exc_4000++;
    }

    printf("Employee number: %d\n", emp_no);
    printf("Overtime Payment: %d\n", overtime_payment);
    printf("-----\n");
}

float percentage_exceeding_4000 = (float)emp_exc_4000 / tot_emp * 100;

printf("Percentage of employees exceeding Rs. 4000: %.2f\n",
percentage_exceeding_4000);

}

```
