

# Programming Fundamentals

## Lab Assignments (2<sup>ND</sup> Semester)

Name: Hamza Kamelen

Roll No: **22F-BSAI-09**

Dept: **Artificial Intelligence**

### LAB 01

**Code:**

```
// BIO DATA
printf("-----Bio Data----- \n");

printf("Hamza Kamelen");
printf("\n 22F-BSAI-09");
printf("\n Student of DUET");
```

**Output:**

```
-----Bio Data-----
Hamza Kamelen
 22F-BSAI-09
 Student of DUET
Process returned 0 (0x0)   execution time : 0.031 s
Press any key to continue.
```

## Code:

```
// Stair Case

printf(":_____ \n");
printf("\t:_____ \n");
printf("\t\t:_____ \n");
printf("\t\t\t:_____ \n");
printf("\t\t\t\t:_____ \n");
printf("\t\t\t\t\t:_____ \n");
```

## Output:

```
:
:_____
:
:_____
:
:_____
:
:_____
:
:_____
:
:_____

Process returned 0 (0x0)   execution time : 0.031 s
Press any key to continue.
```

---

# LAB 02

## Code:

```
/*Task 01:
Using scanf() and printf(),Create a program with the help of format
specifiers to print your marks
Marks of Subject 1 Grad of Subject 1
----- */
int MarksEnglish ,MarksMath,MarksUrdu,Obtained,percentage;
printf("\nMarks of English out of 100:--");
scanf("%d",&MarksEnglish);
printf("Marks of English %d",MarksEnglish);

printf("\nMarks of Math out of 100:--");
scanf("%d",&MarksMath);
printf("\nMarks of Math %d",MarksMath);

printf("\nMarks of Urdu out of 100:--");
scanf("%d",&MarksUrdu);
printf("\nMarks of Urdu %d",MarksUrdu);
Obtained= MarksEnglish+MarksMath+MarksUrdu;
printf("\nObtained Marks %d",Obtained);
percentage= (Obtained*100/300);
printf("\nYour Percentage is %d \n",percentage);
if(percentage > 50){
    printf("Your Grade is A");
}else{
    printf("Your Grade is B");
}
```

## Output:

```
Marks of English out of 100:--20
Marks of English 20
Marks of Math out of 100:--50
Marks of Math 50
Marks of Urdu out of 100:--60
Marks of Urdu 60
Obtained Marks 130
Your Percentage is 43
Your Grade is B
Process returned 0 (0x0)    execution time : 6.683 s
Press any key to continue.
```

## Code:

```
//Task02:
Using scanf() and printf() Create a program that has the following
output: (the price should be taken as input from the user)
//Price of one page:    4.75 Rs
//Price of 15 pages:    71.25 Rs
//Price after discount: 60.00 Rs

int Price;
int FifteenPrice ,DisPrice;
printf("Price of one page:-");

scanf("%d",&Price);

printf("\n Price of one page %d",Price);

FifteenPrice = Price*15;

printf("\n Price of 15 page %d",FifteenPrice);

DisPrice = FifteenPrice - 10%

printf("\n Disc Price %d",DisPrice);
```

## Output:

```
Price of one page:-20

Price of one page 20
Price of 15 page 300
Disc Price 16
```

## Code:

```
//Task 03:  
Write a program that can print a table of 9  
printf("\tMultiplication Table of 9\n\n");  
printf("=====\n");  
int num = 9;  
int i;  
  
printf("Multiplication table of %d:\n", num);  
  
for (i = 1; i <= 10; i++) {  
    printf("%d x %d = %d\n", num, i, num * i);  
}
```

## Output:

```
      Multiplication Table of 9  
  
=====  
Multiplication table of 9:  
9 x 1 = 9  
9 x 2 = 18  
9 x 3 = 27  
9 x 4 = 36  
9 x 5 = 45  
9 x 6 = 54  
9 x 7 = 63  
9 x 8 = 72  
9 x 9 = 81  
9 x 10 = 90
```

## Code:

```
//Task 05:  
Take two integers as input from the user and swap the values.  
  
int num1, num2;  
int swap;  
  
printf("Enter First Number");  
scanf("%d",&num1);  
printf("\nEnter Second Number");  
scanf("%d",&num2);  
printf("Before Swap \n First Int %d , \n Second Int: %d",num1,num2);  
swap = num1;  
num1 = num2;  
num2 = swap;  
  
printf("\n After Swap \n First Int %d , \n Second Int: %d",num1,num2);
```

## Output:

```
Enter First Number2  
  
Enter Second Number4  
Before Swap  
First Int 2 ,  
Second Int: 4  
After Swap  
First Int 4 ,  
Second Int: 2  
Process returned 0 (0x0)    execution time : 3.706 s  
Press any key to continue.
```

## Code:

```
/*Task 06:
Using scanf() and Constant, create a program who takes an input which is
radius of circle and calculate area of it. Area of Circle  $A = \pi r^2$  where  $\pi$ 
is constant (3.1415)*/
printf("-----Area of Circle-----");
int radius, Area;
const Pie = 3.142;
printf("\nPlease Enter Radius");
scanf("%d", &radius);
Area = Pie * (radius * radius);
printf("\nArea == %d", Area);
```

## Output:

```
-----Area of Circle-----
Please Enter Radius20

Area == 1200
Process returned 0 (0x0)    execution time : 2.809 s
Press any key to continue.
```

## Code:

```
// Task 07:
Create a program that takes a decimal number from user and displays the whole
number part and fraction part of the number separately
float Numb;
int whole;
float frac;
printf("Please Write Decimal Number");
scanf("%f", &Numb);
printf("\nYour Number is %f", Numb);
whole = Numb;
printf("\n-----Whole Number is %d", whole);
frac = Numb - whole;
printf("\n-----Fraction Number is %f", frac);
```

## Output:

```
Please Write Decimal Number1.56

Your Number is 1.560000
-----Whole Number is 1
-----Fraction Number is 0.560000
Process returned 0 (0x0)   execution time : 3.594 s
Press any key to continue.
```

---

## LAB 03

### Code:

```
//-----TASK-01: Marksheet-----*

printf("-----SUBJECT MARKSHEET-----");
int English, Urdu, Math, Islamiat, Computer;
printf("\nPlease Enter English Marks");
scanf("\n %d", &English);
printf("\nPlease Enter Urdu Marks");
scanf("\n %d", &Urdu);
printf("\nPlease Enter Math Marks");
scanf("\n %d", &Math);
printf("\nPlease Enter Islamiat Marks");
scanf("\n %d", &Islamiat);
printf("\nPlease Enter Computer Marks");
scanf("\n %d", &Computer);
int total = 500;
int Obtained = English + Urdu + Math + Islamiat + Computer;
float percentage = Obtained * 100 / total;
printf("\n\tEnglish: %d \n\tUrdu: %d \n\tMath: %d
\n\tIslamiat: %d \n\tComputer: %d \n\n\tObtained: %d \n
\n\tTotal: %d \n\n\tPercentage: %f", English, Urdu, Math,
Islamiat, Computer, Obtained, total, percentage);
```



## Output:

```
-----SUBJECT MARKSHEET-----
Please Enter English Marks50

Please Enter Urdu Marks60

Please Enter Math Marks70

Please Enter Islamiat Marks80

Please Enter Computer Marks100

    English: 50
    Urdu: 60
    Math: 70
    Islamiat: 80
    Computer: 100

    Obtained: 360

    Total: 500

    Percentage: 72.000000
Process returned 0 (0x0)   execution time : 11.138 s
Press any key to continue.
```

---

## LAB 05

### Code:

```
// DUET Grading System
Marks Input dena h wo Program marks ko dekh kr Humara grade or CGPA dekhae
int Marks;
printf("\t\tPlease Enter Your PF MARKS And CHECK CGPA");
scanf("%d", &Marks);
printf("\t\tYour PF MARKS is: %d", Marks);
if (Marks >= 85)
{
    if (Marks > 100)
    {
```

```

        printf("\t\t You Are Robot? \n Please Enter Correct Marks");
    }
    else
    {
        printf("\t\tGrade: A+ \n\t\t\t\tCGPA:4.0");
    }
}
else if (Marks >= 80 && Marks <= 84)
{
    printf("\t\tGrade: A \n\t\t\t\tCGPA:3.7");
}
else if (Marks >= 75 && Marks <= 79)
{
    printf("\t\tGrade: B+ \n\t\t\t\tCGPA:3.5");
}
else if (Marks >= 70 && Marks <= 74)
{
    printf("\t\tGrade: B \n\t\t\t\tCGPA:3.0");
}
else if (Marks >= 65 && Marks <= 69)
{
    printf("\t\tGrade: C+ \n\t\t\t\tCGPA:2.5");
}
else if (Marks >= 60 && Marks <= 64)
{
    printf("\t\tGrade: C \n\t\t\t\tCGPA:2.0");
}
else if (Marks >= 55 && Marks <= 59)
{
    printf("\t\tGrade: D+ \n\t\t\t\tCGPA:1.5");
}
else if (Marks >= 50 && Marks <= 54)
{
    printf("\t\tGrade: D \n\t\t\t\tCGPA:1.0");
}
else
{
    if (Marks < 1)
    {
        printf("Invalid.....!");
    }
    else
    {
        printf("\n\t\t\t\tFAIL!!");
    }
}
}

```

## Output:

```
Please Enter Your PF MARKS And CHECK CGPA 94
Your PF MARKS is: 94          Grade: A+
          CGPA:4.0
Process returned 0 (0x0)   execution time : 6.413 s
Press any key to continue.
```

---

# LAB 06

## Code:

```
//Task 01: Write a program to print the table till 10 input taken from user
//=====From For Loop=====
int table, num, answer;
printf("Table of ");
scanf("%d", &table);
for (num = 1; num <= 10; num++)
{
    answer = table * num;
    printf("\n %d x %d = %d", table, num, answer);
}
//=====From While Loop=====
int i = 1, table, answer;
printf("Table of ");
scanf("%d", &table);
while (i <= 10)
{
    answer = table * i;
    printf("\n%d x %d = %d", table, i, answer);
    i++;
}
//=====From Do While Loop=====
int i = 1, table, answer;
printf("Table of ");
scanf("%d", &table);
do
{
    answer = table * i;
    printf("\n%d x %d = %d", table, i, answer);
    i++;
} while (i <= 10);
```

## Output:

```
Table of 15

15 x 1 = 15
15 x 2 = 30
15 x 3 = 45
15 x 4 = 60
15 x 5 = 75
15 x 6 = 90
15 x 7 = 105
15 x 8 = 120
15 x 9 = 135
15 x 10 = 150
Process returned 0 (0x0)   execution time : 1.890 s
Press any key to continue.
```

## Code:

```
// Task 02: Write a program that prints the square of the integer taken
as input, until the square is less than 1000

//=====From For Loop=====

int squareinp, i;

printf("print the number which you want to square till 1000: ");
scanf("%d", &squareinp);
for (i = 1; squareinp <= 1000; i++)
{
    squareinp = squareinp * squareinp;
    if (squareinp >= 1000)
    {
        break;
    }
    printf("\n %d Square is %d", i, squareinp);
}

//=====From While Loop=====

int num;
printf("Enter a Number ");
scanf("%d", &num);
```

```

while (num <= 1000)
{
    num = num * num;
    if (num >= 1000)
    {
        break;
    }
    printf("\n%d", num);
}

//=====From Do While loop=====
int num;
printf("Please Enter Number");
scanf("%d", &num);
do
{
    num *= num;
    if (num >= 1000)
    {
        break;
    }
    printf("\n%d", num);
} while (num <= 1000);

```

## Output:

```

print the number which you want to square till 1000: 2

1 Square is 4
2 Square is 16
3 Square is 256
Process returned 0 (0x0)   execution time : 1.211 s
Press any key to continue.

```

## Code:

```

//Task 04" Write a program that take character input terminate when x
is pressed

//=====From For Loop=====
char userInput;
int i;
printf("Lets Begin the Game\n Note:(If you type x or X
        program terminate)\n ");

```

```

for (i = 1; i <= 10; i++)
{
    printf("Please Enter Any Character: ");
    scanf(" %c", &userInput);
    if (userInput == 'x' || userInput == 'X')
    {
        break;
    }
}

//=====From While loop=====

char Alphabet;
while (Alphabet != 'x')
{

    printf("Please Enter Alphabet");
    scanf(" %c", &Alphabet);
    if (Alphabet == 'x' || Alphabet == 'X')
    {
        break;
    }

    printf("\n Your Alphabet is %c", Alphabet);
}

//=====From Do While loop=====

char Alphabet;

do
{
    printf("\nPlease Enter Alphabet ");
    scanf(" %c", &Alphabet);
    if (Alphabet == 'x' || Alphabet == 'X')
    {
        break;
    }

    printf("\nYour Alphabet is %c", Alphabet);
}while (Alphabet != 'x');

```

## Output:

```
Lets Begin the Game
Note:(If you type x or X program terminate)
Please Enter Any Character: e
Please Enter Any Character: r
Please Enter Any Character: t
Please Enter Any Character: y
Please Enter Any Character: u
Please Enter Any Character: u
Please Enter Any Character: u
Please Enter Any Character: u
Please Enter Any Character: x

Process returned 0 (0x0)    execution time : 8.782 s
Press any key to continue.
```

## Code:

```
// Task 05: Write a program that character input unless Enter is pressed
and print total no of inputs
//=====From For Loop=====
char userInput;
int count = 0;
for (;;)
{
    printf("\nWrite any Character: ");
    userInput = getchar();
    if (userInput == '\n')
    {
        break;
    }
    count++;
}
printf("Your Counting is %d", count);
//=====From While Loop=====
char ch;
int count = 0;
printf("\nPlease Enter Character ");
while (ch != '\n')
{
    ch = getchar();
    if (ch == '\n')
    {
        break;
    }
}
```

```

        count++;
    }
    printf("Your Count is %d", count);
    //=====From Do While Loop=====
    char ch;
    int count = 0;
    printf("\nPlease Enter Character ");
    do
    {
        ch = getchar();
        if (ch == '\n')
        {
            break;
        }
        count++;
    } while (ch != '\n');
    printf("Your Count is %d", count);
}

```

**Output:**

```
Write any Character: sansashjsajsaddsadsahjajsasahjfsafsfadssdd bcnsbcbebcbebcbebhcebcjebcjebjcbebc
Your Counting is 77
Process returned 0 (0x0)   execution time : 7.928 s
Press any key to continue.
```

**Code:**

```
// Task:05 Write a Series of first 50 odd and even numbers

//=====From For Loop=====

for (i = 1; i <= 50; i++)
{
    if (i % 2 == 0)
    {
        printf("\nThis number is even %d", i);
    }
}
```



```

        else
        {
            printf("\nThis number is odd %d", i);
        }

    }

//=====From While Loop=====

int i = 1;
while (i != 50)
{
    if (i % 2 == 0)
    {
        printf("\nThis number is even %d", i);
        i++;
    }
    else
    {
        printf("\nThis number is odd %d", i);
        i++;
    }
}

//=====Do While Loop=====

int i = 1;
do
{
    if (i % 2 == 0)
    {
        printf("\nThis number is even %d", i);
        i++;
    }
    else
    {
        printf("\nThis number is odd %d", i);
        i++;
    }
} while (i != 50);

```

## Output:

```
This number is even 22
This number is odd 23
This number is even 24
This number is odd 25
This number is even 26
This number is odd 27
This number is even 28
This number is odd 29
This number is even 30
This number is odd 31
This number is even 32
This number is odd 33
This number is even 34
This number is odd 35
This number is even 36
This number is odd 37
This number is even 38
This number is odd 39
This number is even 40
This number is odd 41
This number is even 42
This number is odd 43
This number is even 44
This number is odd 45
This number is even 46
This number is odd 47
This number is even 48
This number is odd 49
This number is even 50
Process returned 0 (0x0)    execution time : 0.057 s
Press any key to continue.
```

## Code:

```
/*Task 06(1): Print the following series:
1. 1,2, 3...30*/
//=====From For Loop=====
int i;
for (i = 1; i <= 30; i++)
{
    printf("\n%d", i);
}
```

```

//=====From While Loop=====
int i = 1;
while (i <= 30)
{
    printf("\n%d", i);
    i++;
}

//=====Do While Loop=====

int i = 1;
do
{
    printf("\n%d", i);
    i++;
} while (i <= 30);

```

## Output:

```

1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30,
Process returned 0 (0x0)   execution time : 0.030 s
Press any key to continue.

```

## Code:

```

// Task06(2): Print the following series:

2.  1,2,2,3,3,4,4, 5...50

//=====From For Loop=====
int i = 1;
printf("%d,", i);
i++;
for (i; i <= 50; i++)
{
    printf("%d,", i);
    printf("%d,", i);
}

```

```

//=====From While Loop=====
int i = 1;
printf("%d", i);
i++;
while (i <= 50)

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

//=====Do While Loop=====
int i = 1;
printf("%d", i);
i++;
do
{
    printf("\n%d", i);
    printf("\n%d", i);
    i++;
} while (i <= 50);

```

## Output:

```

1,2,2,3,3,4,4,5,5,6,6,7,7,8,8,9,9,10,10,11,11,12,12,13,13,14,14,15,15,16,16,17,17,18,18,19,19,20,20,21,21,22,22,23,23,24,24,25,25,26,26,27,27,28,28,29,29,30,30,31,31,32,32,33,33,34,34,35,35,36,36,37,37,38,38,39,39,40,40,41,41,42,42,43,43,44,44,45,45,46,46,47,47,48,48,49,49,50,50,
Process returned 0 (0x0)   execution time : 0.028 s
Press any key to continue.

```

## Code:

```

// Task06(3): Print the following series: Fibonacci series
3.    0,1,1,2,3,5,8...100

// From For Loop
int n, first = 0, second = 1, next, i;
printf("No of Terms: ");
scanf("%d", &n);
printf("%d, %d, ", first, second);

```

```

// i=3 bcz first two terms written Already
for (i = 3; i <= n; i++)
{
    next = first + second;
    if (next <= 100)
    {
        break;
    }
    printf("%d, ", next);
    first = second;
    second = next;
}

// From While Loop
int n, first = 0, second = 1, next, i;
printf("No of Terms: ");
scanf("%d", &n);
printf("%d, %d, ", first, second);
i = 3; // bcz first two terms written Already
while (i <= n)
{
    next = first + second;
    if (next <= 100)
    {
        break;
    }
    printf("%d, ", next);
    first = second;
    second = next;
    i++;
}

// From Do While Loop
int n, first = 0, second = 1, next, i;
printf("No of Terms: ");
scanf("%d", &n);
printf("%d, %d, ", first, second);
i = 3; // bcz first two terms written Already
do
{
    next = first + second;
    if (next <= 100)
    { break;
    }
    printf("%d, ", next);
    first = second;
    second = next;
    i++;
} while (i <= n);

```

## Output:

```
No of Terms: 2
0, 1,
Process returned 0 (0x0)   execution time : 0.556 s
Press any key to continue.
```

---

## LAB 07

### Code:

```
/*Task 01:
Write a C Program to print half pyramid as using numbers as shown in
figure below.
1
1 2
1 2 3
1 2 3 4
1 2 3 4 5 */

// From For Loop
int rows, i, j;
printf("Enter Rows: ");
scanf("%d", &rows);
for (i = 1; i <= rows; i++)
{
    for (j = 1; j <= i; j++)
    {
        printf("%d ", j);
    }
    printf("\n");
}

// From While Loop
int rows, i, j;
printf("Enter Rows: ");
scanf("%d", &rows);
i = 1;
```

```
while (i <= rows)
{
    j = 1;
    while (j <= i)
    {
        printf("%d ", j);
        j++;
    }
    i++;
    printf("\n");
}

// From Do While Loop
int rows, i, j;
printf("Enter Rows: ");
scanf("%d", &rows);
i = 1;
do
{
    j = 1;
    do
    {
        printf("%d ", j);
        j++;
    } while (j <= i);
    i++;
    printf("\n");
} while (i <= rows);
```

## Output:

```
Enter Rows: 6
1
1 2
1 2 3
1 2 3 4
1 2 3 4 5
1 2 3 4 5 6
```

## Code:

```
/*Task 02:
Write a C program to print square star (*) pattern series of n rows
For Example: if n=5 the star pattern should be printed like:

*****
*****
*****
*****
***** */

// From For Loop
int rows, i, j;
printf("Enter Rows: ");
scanf("%d", &rows);
for (i = 1; i <= rows; i++)
{
    for (j = 1; j <= rows; j++)
    {
        printf("* ");
    }
    printf("\n");
}

// From While Loop
int rows, i, j;
printf("Enter Rows: ");
scanf("%d", &rows);
i = 1;
while (i <= rows)
{
    j = 1;
    while (j <= rows)
    {
        printf("* ");
        j++;
    }
    i++;
    printf("\n");
}

// From Do While Loop
int rows, i, j;
printf("Enter Rows: ");
scanf("%d", &rows);
i = 1;
```



```

do
{
    j = 1;

    do
    {
        printf("*");
        j++;
    } while (j <= rows);
    i++;
    printf("\n");
} while (i <= rows);

```

## Output:

```

Enter Rows: 9
* * * * *
* * * * *
* * * * *
* * * * *
* * * * *
* * * * *
* * * * *
* * * * *
* * * * *

```

## Code:

```

//Task 3: Write a C program to print Fibonacci series up to n terms.

// From For Loop
int n, first = 0, second = 1, next, i;

printf("No of Terms: ");
scanf("%d", &n);
printf("%d, %d, ", first, second);
// i=3 bcz first two terms written Already

```

```

for (i = 3; i <= n; i++)
{
    next = first + second;
    printf("%d, ", next);
    first = second;
    second = next;
}
// From While Loop
int n, first = 0, second = 1, next, i;
printf("No of Terms: ");
scanf("%d", &n);
printf("%d, %d, ", first, second);
i = 3; // bcz first two terms written Already
while (i <= n)
{
    next = first + second;
    printf("%d,", next);
    first = second;
    second = next;
    i++;
}
// From Do While Loop
int n, first = 0, second = 1, next, i;
printf("No of Terms: ");
scanf("%d", &n);
printf("%d, %d, ", first, second);
i = 3; // bcz first two terms written Already
do {
    next = first + second;
    printf("%d,", next);
    first = second;
    second = next;
    i++;
} while (i <= n);

```

## Output:

```

No of Terms: 20
0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, 233, 377, 610, 987, 1597, 2584, 4181,
Process returned 0 (0x0)   execution time : 0.954 s
Press any key to continue.

```

## Code:

```
/*Task 4:  
    Use for-loop to print the multiplication table below:
```

```
1  2  3  4  5  6  7  8  9  
2  4  6  8 10 12 14 16 18  
3  6  9 12 15 18 21 24 27  
4  8 12 16 20 24 28 32 36  
5 10 15 20 25 30 35 40 45  
6 12 18 24 30 36 42 48 54  
7 14 21 28 35 42 49 56 63  
8 16 24 32 40 48 56 64 72  
9 18 27 36 45 54 63 72 81*/
```

```
// From For Loop
```

```
int rows = 9, cols = 9, i, j;  
printf("Multiplication Table: \n");  
for (i = 1; i <= rows; i++)  
{  
    for (j = 1; j <= cols; j++)  
    {  
        printf("%d\t", i * j);  
    }  
    printf("\n");  
}
```

```
// From While Loop
```

```
int rows = 9, cols = 9, i, j;  
printf("Multiplication Table: \n");  
i = 1;  
  
while (i <= rows)  
{  
    j = 1;  
    while (j <= cols)  
    {  
        printf("%d\t", i * j);  
        j++;  
    }  
    i++;  
    printf("\n");  
}
```

```
// From Do While Loop
int rows = 9, cols = 9, i, j;
printf("Multiplication Table: \n");
i = 1;
do
{
    j = 1;
    do
    {
        printf("%d\t", i * j);
        j++;
    } while (j <= cols);
    i++;
    printf("\n");
} while (i <= rows);
```

## Output:

```
Multiplication Table:
1      2      3      4      5      6      7      8      9
2      4      6      8     10     12     14     16     18
3      6      9     12     15     18     21     24     27
4      8     12     16     20     24     28     32     36
5     10     15     20     25     30     35     40     45
6     12     18     24     30     36     42     48     54
7     14     21     28     35     42     49     56     63
8     16     24     32     40     48     56     64     72
9     18     27     36     45     54     63     72     81
```

## Code:

```
//Task 06: Write a program to print the table till 10 input taken from user

//=====From For Loop=====
int table, num, answer;

printf("Table of ");
scanf("%d", &table);
```

```

for (num = 1; num <= 10; num++)
{
    answer = table * num;
    printf("\n %d x %d = %d", table, num, answer);
}
//=====From While Loop=====
int i = 1, table, answer;
printf("Table of ");
scanf("%d", &table);
while (i <= 10)
{
    answer = table * i;
    printf("\n%d x %d = %d", table, i, answer);
    i++;
}
//=====From Do While Loop=====
int i = 1, table, answer;
printf("Table of ");
scanf("%d", &table);
do
{
    answer = table * i;
    printf("\n%d x %d = %d", table, i, answer);
    i++;
} while (i <= 10);

```

## Output:

```

Table of 50

50 x 1 = 50
50 x 2 = 100
50 x 3 = 150
50 x 4 = 200
50 x 5 = 250
50 x 6 = 300
50 x 7 = 350
50 x 8 = 400
50 x 9 = 450
50 x 10 = 500

```

---

# LAB 08

## Code:

```
/*Task 1: An electric power distribution company charges its domestic consumers as follows.
```

Consumption Units	Rate of Charge
0-200	Rs.0.50 per unit
201-400	Rs.100 plus Rs.0.65 per unit excess 200
401-600	Rs.230 plus Rs.0.80 per unit excess of 400.

```
Write a C program that reads the customer number and power consumed and prints the amount to be paid by the customer. The program runs at least once.
*/
```

```
int customerNo, powerConsumed;
float amountToPaid;
printf("=== Welcome ===");

do
{
    printf("\nEnter Your Customer Number ");
    scanf("%d", &customerNo);

    printf("Enter Power Consumed: ");
    scanf("%d", &powerConsumed);

    printf("Customer No: %d\nAmount to be paid: %d\n",
        customerNo, powerConsumed);

    if (powerConsumed <= 200)
    {
        amountToPaid = powerConsumed * 0.50;
    }
    else if (200 < powerConsumed <= 400)
    {
        amountToPaid = 100 + (powerConsumed - 200) * 0.65;
    }
    else if (400 < powerConsumed <= 600)
    {
        amountToPaid = 230 + (powerConsumed - 400) * 0.80;
    }
    else
    {
        amountToPaid = 0;
    }
}
```

```

    if (amountToPaid != 0)
    {
        printf("Customer Number: %d\n", customerNo);
        printf("Amount to be paid: Rs. %.2f\n", amountToPaid);
    }
    else
    {
        printf("Invalid power consumption.\n");
    }
    printf("=====");
} while (1);

```

## Output:

```

=== Welcome ===
Enter Your Customer Number 12345
Enter Power Consumed: 123
Customer No: 12345
:Power Consumed 123
Customer Number: 12345
Amount to be paid: Rs. 61.50
=====
Enter Your Customer Number

```

## Code:

```

/*Task 2:
Write a C program that allows the user to enter in 5 grades, ie, marks
between 0 - 100. The program must calculate the average mark, and state
the number of marks less than 65.*/
int i = 1, marksofSubject[i], total, fail = 0;
float avg;

printf("Enter 5 Subject Marks (0-100)");
printf("\nYou will Fail if you got Less than 65 Marks:");

```

```

do
{
    printf("\nEnter %d Subject Marks: ", i);
    scanf("%d", &marksofSubject[i]);
    if (marksofSubject[i] > 100)
    {
        printf("Please Enter Valid Marks");
        break;
    }
    total += marksofSubject[i];

    if (marksofSubject[i] < 65)
    {
        fail++;
    }
    i++;
} while (i <= 5);

if (fail == 0)
{
    printf("\nCongratulations You Passed All Subjects ");
}
else
{
    printf("\nOh! Failed in %d Subjects ", fail);
}
avg = total / 5;
printf("\nYour Total Marks is %d", total);
printf("\nYour Average is %.2f ", avg);

```

## Output:

```

Enter 5 Subject Marks (0-100)
You will Fail if you got Less than 65 Marks:
Enter 1 Subject Marks: 55

Enter 2 Subject Marks: 65

Enter 3 Subject Marks: 100

Enter 4 Subject Marks: 63

Enter 5 Subject Marks: 15

Oh! Failed in 3 Subjects
Your Total Marks is 298
Your Average is 59.00

```



## Code:

```
/*Task 3:
Write a program that will generate even numbers and calculate sum and average
of those numbers.*/
int sum, i = 1;
float avg;
do
{
    if (i % 2 == 0)
    {
        printf("Even Number %d\n", i);
        sum += i;
    }
    i++;
} while (i <= 300);
avg = sum / i;
printf("Sum: %d\n", sum);
printf("Average: %f", avg);
printf("\n");
```

## Output:

```
Even Number 266
Even Number 268
Even Number 270
Even Number 272
Even Number 274
Even Number 276
Even Number 278
Even Number 280
Even Number 282
Even Number 284
Even Number 286
Even Number 288
Even Number 290
Even Number 292
Even Number 294
Even Number 296
Even Number 298
Even Number 300
Sum: 22650
Average: 75.000000
```

## Code:

```
/*Task 4:
Using do while loop, program to show a below multiplication table:
    1   2   3   4   5   6   7   8   9
    2   4   6   8  10  12  14  16  18
    3   6   9  12  15  18  21  24  27
    4   8  12  16  20  24  28  32  36
    5  10  15  20  25  30  35  40  45
    6  12  18  24  30  36  42  48  54
    7  14  21  28  35  42  49  56  63
    8  16  24  32  40  48  56  64  72
    9  18  27  36  45  54  63  72  81 */

printf("Mutiplication Table\n");
int row = 1, col;

do
{
    col = 1;
    do
    {
        printf("%d\t", row * col);
        col++;
    } while (col <= 9);
    printf("\n");
    row++;
} while (row <= 9);
```

## Output:

```
Mutiplication Table
1      2      3      4      5      6      7      8      9
2      4      6      8     10     12     14     16     18
3      6      9     12     15     18     21     24     27
4      8     12     16     20     24     28     32     36
5     10     15     20     25     30     35     40     45
6     12     18     24     30     36     42     48     54
7     14     21     28     35     42     49     56     63
8     16     24     32     40     48     56     64     72
9     18     27     36     45     54     63     72     81
```

## Code:

```
/*Task 5:
Write a program that will generate years from 1950 to 2015 and mention if the
year is leap year. */
/*Notes:
A year that is divisible by 4 is a leap year.
However, years divisible by 100 are not leap years, unless...
The year is also divisible by 400, in which case it is a leap year.*/
int year = 1950;
do
{
    if ((year % 4 == 0 && year % 100 != 0) || year % 400 == 0)
    {
        printf("%d is a Leap Year\n", year);
    }
    year++;
} while (year <= 2015);
```

## Output:

```
1952 is a Leap Year
1956 is a Leap Year
1960 is a Leap Year
1964 is a Leap Year
1968 is a Leap Year
1972 is a Leap Year
1976 is a Leap Year
1980 is a Leap Year
1984 is a Leap Year
1988 is a Leap Year
1992 is a Leap Year
1996 is a Leap Year
2000 is a Leap Year
2004 is a Leap Year
2008 is a Leap Year
2012 is a Leap Year
```

## Code:

```
/*Task 6:
Write a program to output a table of values of the integers starting at 1 and
their squares. Label the table at the top of the columns. For example, your
output might look like this:

                Number   Square
                1         1
                3         9*/

int i = 1;
do
{
    printf("Number: %d\nSquare: %d\n -----\n", i, i * i);
    i++;
} while (i <= 50);
```

## Output:

```
-----
Number: 40
Square: 1600
-----
Number: 41
Square: 1681
-----
Number: 42
Square: 1764
-----
Number: 43
Square: 1849
-----
Number: 44
Square: 1936
-----
Number: 45
Square: 2025
-----
Number: 46
Square: 2116
-----
Number: 47
Square: 2209
-----
Number: 48
Square: 2304
-----
Number: 49
Square: 2401
-----
Number: 50
Square: 2500
-----
```

## Code:

```
/*Task 7:
Write a program to print all Prime numbers between 1 to 500.
A Prime Number can be divided evenly only by 1, or itself. And it must be a
whole number greater than 1.
Example:
5 can only be divided evenly by 1 or 5, so it is a prime number. */

// From For Loop
int count, i, n;
for (n = 1; n <= 500; n++)
{
    count = 0;
    for (i = 1; i <= n; i++)
    {
        if (n % i == 0)
        {
            count++;
        }
    }
    if (count == 2)
    {
        printf("%d is a Prime Number\n", n);
    }
}

// From While loop
int count, n, i;
n = 1;
while (n <= 500)
{
    i = 1;
    count = 0;
    while (i <= n)
    {
        if (n % i == 0)
        {
            count++;
        }
        i++;
    }
    if (count == 2)
    {
        printf("%d is Prime number\n", n);
    }
    n++;
}
```

```
// From Do While
int count, n, i;
n = 1;
do
{
    i = 1;
    count = 0;
    do
    {
        if (n % i == 0)
        {
            count++;
        }
        i++;
    } while (i <= n);
    if (count == 2)
    {
        printf("%d is a prime Number\n", n);
    }
    n++;
} while (n <= 500);
```

## Output:

```
349 is a Prime Number
353 is a Prime Number
359 is a Prime Number
367 is a Prime Number
373 is a Prime Number
379 is a Prime Number
383 is a Prime Number
389 is a Prime Number
397 is a Prime Number
401 is a Prime Number
409 is a Prime Number
419 is a Prime Number
421 is a Prime Number
431 is a Prime Number
433 is a Prime Number
439 is a Prime Number
443 is a Prime Number
449 is a Prime Number
457 is a Prime Number
461 is a Prime Number
463 is a Prime Number
467 is a Prime Number
479 is a Prime Number
487 is a Prime Number
491 is a Prime Number
499 is a Prime Number
```

## Code:

```
/*Task 8:
Write a program to enter any number and calculate its factorial using do-while.*/
int factorial, i = 1;
printf("Please Enter a Factorial Number ");
scanf("%d", &factorial);
int given = factorial;
factorial = 1;
do
{
    factorial *= i;
    i++;
} while (i <= given);
printf("Factorial of %d is %d", given, factorial);
```

## Output:

```
Please Enter a Factorial Number 5
Factorial of 5 is 120
Process returned 0 (0x0)   execution time = 0.000 sec
Press any key to continue.
```

## Code:

```
/*Task 9:
Write a C program to enter any number from user and find the reverse of
number, the program executes once and if user wants to repeat the program he
will press Y.
Example:
Input:1234
Output:4321*/
// From For loop
int num, r;
printf("Please Enter Number ");
scanf("%d", &num);
for (; num > 0;)
{
    r = num % 10;
```

```

        printf("%d", r);
        num /= 10;
    }

// From While Loop

int num, r;

printf("Please Enter Number ");
scanf("%d", &num);
while (num > 0)
{
    r = num % 10;
    printf("%d", r);
    num /= 10;
}

// From Do While Loop

int num, r;

printf("Please Enter Number ");
scanf("%d", &num);
do
{
    r = num % 10;
    printf("%d", r);
    num /= 10;
} while (num > 0);

```

## Output:

```

Please Enter Number 5236
6325
Process returned 0 (0x0)
Press any key to continue.

```



## Code:

```
/*Task 10:
Write a do while loop which will produce the following output.
1
22
333
4444
55555*/
int num = 1;
int count = 1;
do
{
    int i = 1;

    do
    {
        printf("%d", num);
        i++;
    } while (i <= count);

    printf("\n");
    num++;
    count++;
} while (num <= 5);
```

## Output:

```
1
22
333
4444
55555

Process returned 0 (0x0)   execution
Press any key to continue.
```

---

# LAB 09

## Code:

```
//Task 01:  
Write a program that will insert 10 int in array and print in new lines  
int arr[10], i;  
printf("Please Enter 10 Numbers \n");  
for (i = 0; i < 10; i++)  
{  
    scanf("\n%d", &arr[i]);  
    printf("\nvalue %d: %d\n", i + 1, arr[i]);  
}
```

## Output:

```
Please Enter 10 Numbers  
1  
  
value 1: 1  
2  
  
value 2: 2  
3  
  
value 3: 3  
4  
  
value 4: 4  
5  
  
value 5: 5  
6  
  
value 6: 6  
7  
  
value 7: 7  
8  
  
value 8: 8  
12  
  
value 9: 12  
6  
  
value 10: 6
```

## Code:

```
//Task 2:  
Write a program that will ask user to input index number from where  
to delete a number from array
```

```
int arr[5], delete;  
printf("You have to input 5 values for array\n");  
  
for (int i = 0; i < 5; i++)  
{  
    scanf("%d", &arr[i]);  
}  
  
printf("Array created successful");  
printf("\nPlease Enter which index do you want to delete");  
scanf("%d", &delete);  
arr[delete] = 0;  
  
for (int h = 0; h < 5; h++)  
{  
    printf("%d", arr[h]);  
}
```

## Output:

```
You have to input 5 values for array  
5  
1  
5  
4  
2  
Array created successful  
Please Enter which index do you want to delete2  
51042  
Process returned 0 (0x0)    execution time : 21.38  
Press any key to continue.
```

## Code:

```
//Task 3:  
Write a program that will find largest and smallest number from Array  
  
int a[5] = {1, 2, 5, 9, 3};  
int min, max;  
min = a[1];  
max = a[1];  
  
for (int i = 0; i < 5; i++)  
{  
    if (min > a[i])  
    {  
        min = a[i];  
    }  
    if (max < a[i])  
    {  
        max = a[i];  
    }  
}  
  
printf("Minimum %d", min);  
printf("Maximum %d", max);
```

## Output:

```
Minimum 1  
Maximum 9  
Process returned 0 (0x0)  
Press any key to continue.
```

## Code:

```
//Task 4:
Take 10 numbers from user in an array calculate the sum avg of those num

int arr[10], sum = 0, i, avg;
printf("Please Enter 10 Numbers \n");

for (i = 0; i < 10; i++)
{
    scanf("\n%d", &arr[i]);
    sum += arr[i];
}

avg = sum / i;
printf("Average: %d\nSum: %d", avg, sum);
```

## Output:

```
Please Enter 10 Numbers
5
6
4
12
3
4
5
6
2
2
Average: 4
Sum: 49
Process returned 0 (0x0)
Press any key to continue.
```

## Code:

```
//Array Reverse
/*Write a program to create two character arrays of same length and
copy the content of one array into another in reverse order.*/
char First[5] = {'H', 'A', 'M', 'Z', 'A'};
char second[5];
int i, j = 0, k;
for (i = 4; i >= 0; i--)
{
    printf(" %c", First[i]);
    second[j] = First[i];
    j++;
}
printf("\n");
for (k = 0; k < 5; k++)
{
    printf(" %c", second[k]);
}
```

## Output:

```
A Z M A H
Process returned 0 (0x0)
Press any key to continue.
```

## Code:

```
// Bubble Sorting
// Ascending

int arr[5] = {22, 9, 3, 5, 8};
int temp, i, n;
for (i = 0; i < 5; i++)
{
    for (n = 0; n < 5; n++)
    {
        if (arr[n] < arr[n + 1])
        {
```

```

        temp = arr[n];
        arr[n] = arr[n + 1];
        arr[n + 1] = temp;
    }
}
}
int h;
for (h = 0; h < 5; h++)
{
    printf("\n%d", arr[h]);
}

```

## Output:

```

22
9
8
5
3
Process returned 0 (0x0)
Press any key to continue.

```

## Code:

```

// Bubble Sorting
// Descending

int arr[5] = {22, 9, 3, 5, 8};
int temp, i, n;
for (i = 0; i < 5; i++)
{
    for (n = 0; n < 5; n++)
    {
        if (arr[n] > arr[n + 1])
        {
            temp = arr[n];
            arr[n] = arr[n + 1];
            arr[n + 1] = temp;
        }
    }
}

```

```
int h;  
for (h = 0; h < 5; h++)  
{  
    printf("\n%d", arr[h]);  
}
```

## Output:

```
0  
3  
5  
8  
9  
Process returned 0 (0x0)   0  
Press any key to continue.
```

---

# LAB 10

## Code:

```
/*Task 1:  
Write a program that accepts one string and then find the length of  
That string and print it.*/  
  
char name[20];  
int length;  
printf("Please Enter Your Name: ");  
gets(name);  
printf("\nYour Name is ");  
puts(name);  
  
// Checking Length & Print Length  
length = strlen(name);  
printf("%d", length);
```



## Output:

```
Please Enter Your Name: Hamza

Your Name is Hamza
Length is: 5
Process returned 0 (0x0)   execution
Press any key to continue.
```

## Code:

```
/*Task 2:
Write a program that takes input in a string (char array) and 'find'
a particular character and 'replace' that character. The program
should replace all the occurrences of that particular character with the
new character.
For example:
Array: "I love programming" find character: 'm'
replace with character: 'n'
then the string should become "I love progranning"*/
char name[20];
char old;
char replace;
printf("Please Enter Your Name: ");
gets(name);
printf("\nYour Name is ");
puts(name);
printf("What you want to replace? ");
scanf("%c", &old);
printf("Replace With? ");
scanf("%c", &replace);
for (int i = 0; i < strlen(name); i++)
{
    if (name[i] == old)
    {
        name[i] = replace;
    }
}
puts(name);
```

## Output:

```
Please Enter Your Name: Hamzaa

Your Name is Hamzaa
What you want to replace? a
Replace With? z
Hzmzzz

Process returned 0 (0x0)   execution
Press any key to continue.
```

## Code:

```
/*Task 3:
Fifteen numbers are entered from the keyboard into an array with 3 rows
and 5 columns. The number to be searched is entered through the keyboard
by the user. Write a program to find if the number to be searched is
present in the array and if it is present, display the number of times
it appears in the array. */
int arr[3][5];
int i, j, value, count = 0;
// Taking Value from User
for (i = 0; i < 3; i++)
{
    for (j = 0; j < 5; j++)
    {
        printf("Enter the number at array[%d][%d] ", i, j);
        scanf("%d", &arr[i][j]);
    }
}
// Search value
printf("Please Enter number to be searched: ");
scanf("%d", &value);
// Count Search Value
for (i = 0; i < 3; i++)
{
    for (j = 0; j < 5; j++)
    {
        if (arr[i][j] == value)
        {
            count++;
        }
    }
}
```

```

    }
}
printf("The number %d appeared %d times in Array ", value, count);

```

## Output:

```

Enter the number at array[0][0] 2
Enter the number at array[0][1] 3
Enter the number at array[0][2] 4
Enter the number at array[0][3] 5
Enter the number at array[0][4] 6
Enter the number at array[1][0] 1
Enter the number at array[1][1] 2
Enter the number at array[1][2] 5
Enter the number at array[1][3] 66
Enter the number at array[1][4] 1
Enter the number at array[2][0] 2
Enter the number at array[2][1] 2
Enter the number at array[2][2] 2
Enter the number at array[2][3] 3
Enter the number at array[2][4] 4
Please Enter number to be searched: 5
The number 5 appeared 2 times in Array
Process returned 0 (0x0)    execution time :
Press any key to continue.

```

## Code:

```

/*Task 4:
Write a program to pick up the largest number and sum of all the
integers from any 5 x 5 matrix. */
int matrix[5][5], i, j;
int largest = 0, sum = 0;
// Taking Value from User
for (i = 0; i < 5; i++)
{
    for (j = 0; j < 5; j++)
    {
        printf("Enter the number at matrix[%d][%d] ", i, j);
        scanf("%d", &matrix[i][j]);
    }
}

```

```

// Largest Number
for (i = 0; i < 5; i++)
{
    for (j = 0; j < 5; j++)
    {
        if (matrix[i][j] > largest)
        {
            largest = matrix[i][j];
        }
    }
}
// Sum of Numbers
for (i = 0; i < 5; i++)
{
    for (j = 0; j < 5; j++)
    {
        sum += matrix[i][j];
    }
}
printf("\nLargest Number %d", largest);
printf("\nSum %d", sum);

```

## Output:

```

Enter the number at matrix[0][0] 1
Enter the number at matrix[0][1] 2
Enter the number at matrix[0][2] 3
Enter the number at matrix[0][3] 4
Enter the number at matrix[0][4] 5
Enter the number at matrix[1][0] 6
Enter the number at matrix[1][1] 1
Enter the number at matrix[1][2] 5
Enter the number at matrix[1][3] 8
Enter the number at matrix[1][4] 9
Enter the number at matrix[2][0] 2
Enter the number at matrix[2][1] 1
Enter the number at matrix[2][2] 4
Enter the number at matrix[2][3] 65
Enter the number at matrix[2][4] 4
Enter the number at matrix[3][0] 5
Enter the number at matrix[3][1] 32
Enter the number at matrix[3][2] 1
Enter the number at matrix[3][3] 2
Enter the number at matrix[3][4] 3
Enter the number at matrix[4][0] 4
Enter the number at matrix[4][1] 5
Enter the number at matrix[4][2] 6
Enter the number at matrix[4][3] 5
Enter the number at matrix[4][4] 5

Largest Number 65
Sum 188
Process returned 0 (0x0)    execution time : 17.580 s
Press any key to continue.

```

## Code:

```
/*Task 5:
Write a program to obtain transpose of a 4x4 matrix. The transpose of
a matrix is obtained by exchanging the elements of each row with
the elements of the corresponding column*/
int matrix[4][4];
int transpose[4][4];
int i, j;
// Taking Value from User
for (i = 0; i < 4; i++)
{
    for (j = 0; j < 4; j++)
    {
        printf("Enter the number at matrix[%d][%d] ", i + 1, j + 1);
        scanf("%d", &matrix[i][j]);
    }
}
// Transpose of a Matrix
for (i = 0; i < 4; i++)
{
    for (j = 0; j < 4; j++)
    {
        transpose[j][i] = matrix[i][j];
    }
}
// Print Original Matrix
printf("-----Original Matrix-----\n");
for (i = 0; i < 4; i++)
{
    for (j = 0; j < 4; j++)
    {
        printf("  %d", matrix[i][j]);
    }
    printf("\n");
}
// Print Transpose of a Matrix
printf("-----Transpose of a Matrix-----\n");
for (i = 0; i < 4; i++)
{
    for (j = 0; j < 4; j++)
    {
        printf("  %d", transpose[i][j]);
    }
    printf("\n");
}
```

## Output:

```
Enter the number at matrix[1][1] 1
Enter the number at matrix[1][2] 2
Enter the number at matrix[1][3] 4
Enter the number at matrix[1][4] 5
Enter the number at matrix[2][1] 6
Enter the number at matrix[2][2] 8
Enter the number at matrix[2][3] 1
Enter the number at matrix[2][4] 5
Enter the number at matrix[3][1] 3
Enter the number at matrix[3][2] 5
Enter the number at matrix[3][3] 4
Enter the number at matrix[3][4] 2
Enter the number at matrix[4][1] 6
Enter the number at matrix[4][2] 1
Enter the number at matrix[4][3] 2
Enter the number at matrix[4][4] 3
-----Original Matrix-----
  1  2  4  5
  6  8  1  5
  3  5  4  2
  6  1  2  3
-----Transpose of a Matrix-----
  1  6  3  6
  2  8  5  1
  4  1  4  2
  5  5  2  3
```

## Code:

```
/*Task 6:
Write a program to add two 6 x 6 matrices. Add result in third array
and print this array.*/
int result[6][6],i,j;
int firstMatrix[6][6] = {
    {1, 2, 3, 4, 5, 6},
    {2, 4, 6, 8, 9, 10},
    {3, 6, 9, 12, 15, 16},
    {4, 8, 12, 16, 19, 21},
    {5, 10, 15, 20, 25, 30},
    {2, 1, 3, 4, 5, 6} };
int secondMatrix[6][6] = {
    {1, 2, 3, 4, 5, 6},
    {2, 3, 6, 8, 9, 1},
    {3, 4, 9, 2, 5, 6},
    {4, 5, 1, 6, 9, 1},
    {5, 6, 5, 0, 5, 0},
```

```

        {6, 7, 3, 4, 5, 6}
    };
    // Addition of two matrices
    for(i=0;i<6;i++){
        for(j=0;j<6;j++){
            result[i][j]=firstMatrix[i][j]+secondMatrix[i][j];
        }
    }
    // Print Addition
    for(i=0;i<6;i++){
        for(j=0;j<6;j++){
            printf("%d",result[i][j]);
        }
        printf("\n");
    }
}

```

## Output:

```

2      4      6      8      10     12
4      7      12     16     18     11
6      10     18     14     20     22
8      13     13     22     28     22
10     16     20     20     30     30
8      8      6      8      10     12

Process returned 0 (0x0)   execution time : 0
Press any key to continue.

```

## Code:

```

/*Task 7:
Write a program to multiply any two 3 x 3 matrices. Add result in
third array and print this array.*/
int result[3][3],i,j;
int firstMatrix[3][3] = {
    {1, 2, 3},
    {2, 4, 6},
    {3, 6, 9}
};
int secondMatrix[3][3] = {
    {1, 2, 3},
    {2, 3, 6},
    {3, 4, 9}
};

```

```

// Multiplication of two matrices
for(i=0;i<3;i++){
    for(j=0;j<3;j++){
        result[i][j]=0;
        for(int k=0;k<3;k++){
            result[i][j] += firstMatrix[i][k]*secondMatrix[k][j];
        }
    }
}
// Print Multiplication
for(i=0;i<3;i++){
    for(j=0;j<3;j++){
        printf("%d",result[i][j]);
    }
    printf("\n");
}

```

## Output:

```

14      20      42
28      40      84
42      60     126

```

```

Process returned 0 (0x0)
Press any key to continue.

```

## Code:

```

/*Task 8:
In a bitmap file black is represented by 1 and white by 0. Create a
8X8 pattern and initialize using following data:
1 1 1 0 0 1 1 1
1 0 0 0 0 0 1 0
1 1 1 1 1 1 0 0
1 1 1 0 0 0 1 0
1 0 1 0 1 0 1 0
0 0 0 0 0 0 0 0
1 1 1 1 1 1 1 1
0 0 0 0 1 1 1 1 */
int bitmapPattern[8][8]={
    {1,1,1,0,0,1,1,1},
    {1,0,0,0,0,0,1,0},
    {1,1,1,1,1,1,0,0},
    {1,1,1,0,0,0,1,0},
    {1,0,1,0,1,0,1,0},
    {0,0,0,0,0,0,0,0},
    {1,1,1,1,1,1,1,1},
    {0,0,0,0,1,1,1,1}
}

```



```

        {1,1,1,0,0,0,1,0},
        {1,0,1,0,1,0,1,0},
        {0,0,0,0,0,0,0,0},
        {1,1,1,1,1,1,1,1},
        {0,0,0,0,1,1,1,1}
    };
    int i,j;
    for(i=0;i<8;i++){
        for(j=0;j<8;j++){
            printf("%d  ",bitmapPattern[i][j]);
        }
        printf("\n");
    }
}

```

## Output:

```

1  1  1  0  0  1  1  1
1  0  0  0  0  0  1  0
1  1  1  1  1  1  0  0
1  1  1  0  0  0  1  0
1  0  1  0  1  0  1  0
0  0  0  0  0  0  0  0
1  1  1  1  1  1  1  1
0  0  0  0  1  1  1  1

Process returned 0 (0x0)   execution
Press any key to continue.

```

---

# LAB 11

## Code:

```
/*Task 1:
Write a function to calculate the factorial value of any integer entered
through the keyboard*/
#include <stdio.h>
#include <stdlib.h>
int facto(int n)
{
    if (n == 0)
    {
        return 1;
    }
    else
    {
        return n * facto(n - 1);
    }
}
int main()
{
    int n, fact;
    printf("Please Enter Factorial no: ");
    scanf("%d", &n);
    fact = facto(n);
    printf("Factorial of %d is %d", n, fact);
    return 0;
}
```

## Output:

```
Please Enter Factorial no: 5
Factorial of 5 is 120
Process returned 0 (0x0)    execution time :
Press any key to continue.
```

## Code:

```
/*Task 2:
Write a function power (a, b) to calculate the value of 'a' raised to the
power of 'b'.*/
#include <stdio.h>
#include <stdlib.h>

int power(int a, int b)
{
    if (b == 0)
    {
        return 1;
    }
    else if (b == 1)
    {
        return a;
    }
    else
    {
        return a * power(a, b - 1);
    }
}

int main()
{
    int a, b;
    printf("Enter the base: ");
    scanf("%d", &a);
    printf("Enter the exponent: ");
    scanf("%d", &b);
    int value = power(a, b);
    printf("The value of %d raised to the power of %d is %d\n", a, b, value);
    return 0;
}
```

## Output:

```
Enter the base: 5
Enter the exponent: 6
The value of 5 raised to the power of 6 is 15625

Process returned 0 (0x0)    execution time : 3.734
Press any key to continue.
```

## Code:

```
/*Task 3:\nWrite a general-purpose function to convert any given year into its roman equivalent. The following table shows the roman equivalents of decimal numbers:\nDecimal          Roman\n1                i\n100              c\n5                v\n500              d\n10               x\n1000             m\n50               l\nExample:\nRoman equivalent of 1988 is mdcccclxxxviii\nRoman equivalent of 1525 is mdxxv */\n#include <stdio.h>\n#include <stdlib.h>\n\nvoid convertRoman(int Convertyear)\n{\n    while (Convertyear)\n    {\n        if (Convertyear >= 1000)\n        {\n            printf("M");\n            Convertyear = Convertyear - 1000;\n        }\n        else if (Convertyear >= 500)\n        {\n            printf("D");\n            Convertyear = Convertyear - 500;\n        }\n        else if (Convertyear >= 100)\n        {\n            printf("C");\n            Convertyear = Convertyear - 100;\n        }\n        else if (Convertyear >= 50)\n        {\n            printf("L");\n            Convertyear = Convertyear - 50;\n        }\n        else if (Convertyear >= 10)\n        {\n            printf("X");\n            Convertyear = Convertyear - 10;\n        }\n    }\n}
```

```

    }
    else if (Convertyear >= 5)
    {
        printf("V");
        Convertyear = Convertyear - 50;
    }
    else if (Convertyear >= 1)
    {
        printf("I");
        Convertyear = Convertyear - 1;
    }
}
}
int main()
{
    int year;
    printf("Convert Year into Roman");
    // Taking Year from User
    printf("Please Enter a Year ");
    scanf("%d", &year);
    convertRoman(year);

    return 0;
}

```

## Output:

```

Convert Year into RomanPlease Enter a Year 2054
MMLIIII
Process returned 0 (0x0)   execution time : 7.215 s
Press any key to continue.

```

---

# LAB 12

## Code:

```
/*Task 1:
There is a structure called employee that holds information like employee
code, name, date of joining. Write a program to create an array of the
structure and enter some data into it. Then ask the user to enter current
date. Display the names of those employees whose tenure is 3 or more than 3
years according to the given current date. */

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

struct Employee {
    int EmpCode;
    char name[50];
    int YearofJoining;
} employeeData[2];

int main() {
    int currentYear;
    // Taking Employee data
    for (int i = 0; i < 2; i++) {
        // Clear the input buffer before reading the employee name
        while (getchar() != '\n');
        printf("==== Employee %d =====", i+1);

        printf("\nPlease Enter an Employee code: ");
        scanf("%d", &employeeData[i].EmpCode);

        printf("Please Enter Employee Name: ");
        fgets(employeeData[i].name, 50, stdin);
        employeeData[i].name[strlen(employeeData[i].name)-1]=0;

        printf("Please Enter the Year of Joining: ");
        scanf("%d", &employeeData[i].YearofJoining);
        printf("\n");
    }
    system("cls");
    printf("Enter Current Year: ");
    scanf("%d", &currentYear);

    printf("Employee with tenure 3 or more \n");
    for(int i=0;i<2;i++){
        int tenure = currentYear - employeeData[i].YearofJoining;
        if(tenure >= 3){
            printf("Name: %s \n", employeeData[i].name);
        }
    }
}
```

```

        printf("Year of Joining: %d \n",employeeData[i].YearofJoining);
        printf("\n");
    }
}
return 0;
}

```

## Output:

```

Enter the number of employees: 2
===== Employee Data =====
===== Employee 1 =====
Please Enter an Employee code: 12
Please Enter Employee Name: Hamza
Please Enter the Year of Joining: 2003

===== Employee 2 =====
Please Enter an Employee code: 1234
Please Enter Employee Name: talha
Please Enter the Year of Joining: 2022

```

```

Enter Current Year: 2023
Employees with tenure 3 or more:
Name: Hamza
Year of Joining: 2003

Process returned 0 (0x0)   execution time : 58.112 s
Press any key to continue.

```

-----

# Structures(Book Questions)

**Code:**

```
//Define a structure consisting of two floating-point members, called real &
imaginary. Include the tag complex within the definition
struct complex{
    float real;
    float imaginary;
};
```

**Code:**

```
//Declare the variable x1,x2,x3 to be structures of type complex as describe
in the program

struct complex x1,x2,x3;
x2.real = 20.5;
printf("%f",x2.real);
```

**Code:**

```
//Combine Upper Both Question

struct complex{
    float real;
    float imaginary;
}x1,x2,x3;
```

**Code:**

```
//Declare a structure variable x and assign a value 1.3 , -2.2 in real &
imaginary

struct complex x;
x.real =1.3;
x.imaginary = -2.2;
```



```
printf("%f,%f",x.real,x.imaginary);
```

**Code:**

```
// Declare a one dimensional 100 element array called cx whose elements  
are structure of type complex,  
  
struct complex cx[100];
```

**Code:**

```
// Combine the structure definition and the array declaration  
struct complex  
{  
    float real;  
    float imaginary;  
} cx[100];
```

**Code:**

```
/*Define a structure that contains the following three members  
--> an int called won  
--> an int called lost  
--> an float called percentage  
Include the user-defined data type record within the definition  
*/  
  
struct record  
{  
    int won;  
    int lost;  
    float percentage;  
};
```

## Code:

```
/*Define a structure that contain the following two members
--> a 40-element character called name
--> a structure name stats,of type record
Include the user-defined data type team within the definition */

struct team
{
    char name[40];
    struct record stats;
};
```

## Code:

```
/*Declare a t to be a structure variable of type team as in the previous
problem Write a expression of each member & submember of t.*/
struct record
{
    int won;
    int lost;
    float percentage;
};
struct team
{
    char name[40];
    struct record stats;
} t;
t.stats.won = 2;
printf("%d", t.stats.won);
```

## Code:

```
/*Declare a t to be a structure variable of type team as in the
previous problem.Initialize t as follows:
name: Chichago Bears
won:14
lost:2
percentage:87.5
*/

struct record
```

```

{
    int won;
    int lost;
    float percentage;
};
struct team
{
    char name[40];
    struct record stats;
} t;
t.name = "Hamza Kamelen";
t.stats.won = 14;
t.stats.lost = 2;
t.stats.percentage = 87.5;

return 0;
}

```

---

## Pointers(Book Questions)

**Code:**

```

/*Q1:
Write a function that recieve 5 integer & returns the sum,avg & standard
deviation of these numbers. Call this function from main() and print the
result in main()*/

void Values(int numbers[5],int size,int *sum,float *avg,float *var,float
*std){
    *sum=0;
    for(int i=0;i<size;i++){
        *sum+= numbers[i];
    }
    *avg = *sum/size;
    for(int i=0;i<size;i++){
        *var += ((numbers[i]-*avg)*(numbers[i]-*avg));
    }
    *var = *var/size;
    *std = sqrt(*var);
}

int main(){
    int numbers[5];
    int sum;
    float avg;
    float var;

```

```

float std;
for(int i=0;i<5;i++){
    printf("Please Write %d Number ",i+1);
    scanf("%d",&numbers[i]);
}
Values(numbers,5,&sum,&avg,&var,&std);
printf("\nSum: %d",sum);
printf("\nAverage: %.2f",avg);
printf("\nStandard Deviation: %.2f",std);

return 0;
}

```

## Code:

```

/*Q2:
Write a function that recieve marks recieved by a student in 3 subjects &
returns the percentage,avg of these numbers. Call this function from main()
and print the result in main()*/
void DateSheet(int *marks,int size,int *total,float *avg,float *perc){
    *total=0;
    for(int i=0;i<size;i++){
        *total += marks[i];
    }
    *avg = *total/size;
    *perc = (*total *100)/(size*100);
}
int main(){
    int marks[3];
    int total=0;
    float avg=0;
    float perc;
    marks[3]=0;
    for(int i=0;i<3;i++){
        printf("Marks of %d Subject ",i+1);
        scanf("%d",&marks[i]);
    }
    DateSheet(marks,3,&total,&avg,&perc);
    printf("\nObtained Marks: %d",total);
    printf("\nAverage: %.2f",avg);
    printf("\nPercentage: %.2f",perc);

return 0;
}

```

## Code:

```
/*Q3:
Given three variable x,y,z write a function to circularly shift their values
to right. In other words if x=5,
y=8,z=10 after circular shift y=5, z=8, x=10.Call the function with variable
a,b,c to circular shift values*/
void CircularShift(int *x,int *y,int *z){
int temp = *x;
*x =*z;
*z =*y;
*y = temp;
}
int main(){
    int x=5,y=8,z=10;
    CircularShift(&x,&y,&z);
    printf("%d %d %d",x,y,z);
return 0;
}
```

---