VISION DATE : 8/11/'25
PAGE NO.

Name : Patel Anuragkumar S.

Branch : Civil

Roll no. : 25BCL106

Subject : Computer Programming [Assignment-4]

(#) Programs usings Pointers, Arrays, Strings and Arrays of Pointers.

```
1. # include <stdio.h>
   int main() {
     int a[3][3], b[3][3], Sum[3][3], Sub[3][3], mul[3][3];
     int i, j, k;
     printf ("Enter elements of 1st 3x3 matrix:\n");
     for (i=0;i<3 ; i++)
        for (j=0; j<3 ; j++)
           sconf ("%d", &a[i][j]);
     printf("Enter elements of 2nd ~~@~~x3 matrix:\n");
     for (i=0; j<3; i++)
        for(j=0; j<3; j++)
           sconf ("%d", &b[i][j]);
     for (i=0; i<3; i++)
        for (j=0; j<3;j++)
           sum [i][j] = a[i][j] + b[i][j];
           sub [i] [j] = a[i] [j] ~~+~~ - b[i][j];
        }
     for (i=0; i<3;i++) {
        for (j=0; j<3 ; j++) {
           mul[i][j] = 0;
           for (k=0; k<3 ; k++)
   }          mul [i][j] += a[i][j] * b[j][k];
```

```
    }
  }
  print f ("\nSum:\n");
  for (i=0 ; i<3; i++) {
      for (j=0; j<3; j++) print f ("%d", Sum [i][j]);
      print ("\n");
  }
 printf ("\nsubtraction:\n");
  for (i=0; i<3 ; i++) {
      for (j=0; j<3; j++) printf ("%d",sub [i][j]);
     printf ("\n");
  }
  printf ("\nMultipliccation:\n");
  for (i=0; i<3; i++) {
      for (j=0; j<3; j++) printf ("%d", mul [i][j];
     printf ("\n");
  }
  return 0;
}
```

=> <u>Output:-</u>

```
Enter elements of 1st 3x3 matrix : 1, 2, 3, 4, 6, 2, 3, 1, 0
Enter elements of 2nd 3x3 matrix : 9, 2, 3, 6, 4, 8, 7, 9, 2

Sum :             Substraction:     Multiplication:
 2   4   6          0   0   0        6   36  54
 10  10  10        -2   2  -6       24  108  38
 10  10  2         -4  -8  -2       18   18   0
```

2.
```
# include <stdio.h>
int main() {
    int a [4][4] , arr [16], i, j, k = 0 , temp;
    printf ("Enter 4x4 matrix elements:\n");
    for (i=0; i<4; i++)
        for (j=0; j<4 ;j++)
            scanf ("%d", &a [i][j]);
    for (i=0;i<4;i++)
        for (j=0; j<4; j++)
        arr [k++] = a [i][j];

    for (i=0 ;i<16;i++)
        for (j=i+1; j<16; j++)
            if(arr [i] > arr [j]) {
                temp = arr [i];
                arr [i] = arr [j];
                arr [j] = temp;
            }
        printf ("\nsorted array :\n");
          for (i=0;i<16;i++)
              printf ("%d" , arr [i]);
      return 0;
  }
```

⇒ Output:

```
Enter 4x4 matrix elements: 1, 5, 9, 7, 5, 3, 4, 6, 2,
                      8, 2, 5, 8, 4, 5, 6

Stored array : 1 2 2 3 4 4 5 5 5 5 6 6 7
               8 8 9
```

3.

```
# include <stdio.h>
  int main () {
        int a [3] [3], *p , i, j ;
        int max , min ;
        printf ("Enter 3x3 matrix : \n");
        for (i=0 ; i<3 ; i++ )
        for (j=0; j<3;j ++)
            scanf ("%d", &a [i] [j]);
  p= &a [0] [0];
  max = min = *p ;
  for (i=0;i<9 ;i++) {
     if (*(p+i) > max ) max = *(p+i) ;
     if (*(p+i) < min ) min = *(p+i) ;
  }
    printf ("Largest = %d\n", max );
    printf ("Smallest = %d\n", min );

  return 0;
 }
```

⇒ Output:

```
Enter 3x3 matrix: 10, 11, 12
                  5   6   7
                  1   2   3

Largest = 12
Smallest = 1
```

```
4. # include<stdio.h>
     int main() {
         Char books [3][50];
         int i;
         printf ("Enter name of 3 books:\n");
     for(i= 0; i<3; i++) {
         printf ("Enter book %d: ", i+1);
         Scanf (" %[^\n]", books[i]);
     }
       printf ("\n**** Book list ****\n");
     for (i=0 ; i<3 ;i++ ) {
           printf ("Book %d: %s\n", i+1 , book [i]);
     }
     return 0;
     }
```

=> Output:

```
Enter names of 3 Books:              *** Book list ***
Enter book 1 : Physics               Book 1 : physics
Enter book 2: Chemistry              Book 2 : Chemistry
Enter Book 3: Maths                  Book 3: Maths
```

```
5. # include <stdio.h>
   # include <string.h>
   int main() {
     Char name [100], *token, lastName [50];
     printf ("Enter full name: ");
     Scanf ("% [^\n ]", name );
     token = strotok (name, " ");
     while (token != NULL) {
       strcpy (lastName, token);
   token = strtok (NULL, " ");
  }
  token = strtok (name, " ");
  while (token != NULL && strcmp (token, lastName)
         != 0) {
   printf ("%c. ", token [0]);
   token = strtok (NULL, " ");
   }
   printf ("%s", lastName);
   return 0;
  }
```

=> Output :-

```
Enter full Name : Patel Anuragkumar

 P. Anuragkumar
```

Unit 3 : Functions and Recursive Functions

1.
```
#include <stdio.h>
int power (int a, int b) {
    int result = 1;
    for (int i=1; i<=b; i++) {
      result *= a;
    }
    return result;
 }
    int main() {
     int a, b;
    printf (" Enter base (a): ");
    scanf ("%d", &a);
    printf ("Enter exponent(b): ");
    scanf ("%d", &b);
     printf ("%d^%d = %d\n", a, b, power(a,b));

    return 0;
  }
```

=> Output :-

```
Enter base (a) : 5
Enter exponant (b) : 6
 5^6 = 15625
```

2.

```
 # include <stdio.h>
 int isLeap (int year) {
     if ((year % 4 == 0 && year %100!=0) || (year%.400=
     return 1;
   else
     return 0;
  }
 int main() {
     int year;
    printf ("Enter a year : ");
    scanf ("%d", &year);
    if (isLeap (year))
        printf("%d is a leep year \n", year);
    else
        printf("%d is not a leap year \n", year);

    return 0;
 }
```

=> output :- Enter a year : 2025
2025 is not a leap year.

3.
```
# include <stdio.h>
int factorial (intn) {
  if  (n == 0 || n == 1)
       return 1;
  else
   return n * factorial (n-1);
 }
 int main () {
     int num;
   printf ("Enter a number:");
   scanf ("%d", &num);
   if (num<0)
     printf ("factorial of nagative number doesn't
                 exist.\n");
   else
   printf ("factorial of %d = %d\n", num, factorial
             (num) );

     return 0;
 }
```

⇒ Output:-

```
Enter a number:- 521
factorical of 521 = 0
```

4.
```
#include <stdio.h>
~~int main~~ void swap (int x, inty) {
   int temp;
   temp = x;
    x = y
    y = temp ;
 printf ("\nInside swap function:");
 printf (" \n x= %d , y =% d\n", x, y);
}
```

```
int main() {
 int a, b;
 printf ("Enter two numbers:");
 Scanf ("%d %d", &a, &b);
 printf (" \n Before Calling Swap:");
 printf (" \n a = %d, b = %d", a, b,);
 Swap (a, b);
 printf ("\nAfter calling swap");
 printf ("\na = %d, b = %d\n", a, b);

 return 0;
}
```

⇒ Output:-

```
 Enter two numbers :- 56, 44
Before calling Swap :
 a = 56, b = 44
Inside Swap function :
 x = 44 ,  y = 56
After calling swap ~~fun~~:
a = 56,  b = 44
```

5.
```
#include <Stdio.h>
void findMaxMin (int arr [], int n, int *max, int *min) {
     *max = arr [0];
     *min = arr [0];
     for (int i = 1; i < n; i++) {
        if (arr [i] > *max)
            *max = arr [i];
        if (arr [i] < *min)
            *min = arr [i];
     }
 }

 int main () {
```

```
int arr [50], n, max, min;
printf ("Enter number of elements: ");
scanf ("%d", &n);
printf ("Enter %d elements :\n", n);
for (int i= 0 ; i<n ; i++)
 scanf ("%d" , &arr [i]);
find MaxMin (arr, n, &max, &min);
printf ("\nMaximum value = %d ", max);
printf ("\n Minimum value = %d \n", min);

return 0;
}
```

→ Output:-

```
Enter number of elements: 4
Enter 4 elements:
22
54
96
12
Maximum value = 96
Minimum value = 12
```

6.

```
#include <stdio.h>
float add (float a, float b) {
  return a+b;
}
float subtract (float a, float b) {
   return a-b;
float multiply (float a, float b) {
   return a* b;
float divide ( Float a, float b) {
 if (b == 0) {
  printf ("Error! Division by zero is not allowed \n");
```

```
    return 0;
  }
  return a/b;

int main()
    float num1, num2, result;
   int choise;
    printf("Enter first number: ");
   scanf("%f", &num1);
   printf("Enter second number: ");
    scanf("%f", &num2);
    printf("\nChoose operation: \n");
    printf("1. Add\n");
    printf("2. Subtract \n");
    printf("3. Multiply \n");
    printf("4. Divide\n");
    printf(" Enter your choice: ");
   scanf("%d", &choice);
switch(choice) {
    Case 1:
            result = add(num1, num2);
            printf("Result: %.2f\n", result);
            breeak;
    Case 2:
            result = subtract(num1, num2);
            printf(" Result: %.2f\n", result);
    Ca. break;
    Case 3:
        result = multiply(num1, num2);
         printf("Result : %.2f\n", result);
        breeak;
    case 4:
     result = divide(num1, num2);
```

```
if (num2 != 0)
    printf (" Result: %.2f\n", result);
  break
  default :
     printf ("Invalid choice.\n");
 }
 return 0;
}
```

=> Output:-

```
Enter first number: 56
Enter Second number: 23
choose operation:
 1. Add
 2. Subtract
 3. Multiply
 4. Divide
 Enter your choice :4
 Result : 2.43
```

7.

```
# include <stdio.h>
int main () {
     int a [10] [10], b[10][10], Sum [10][10], Sub[10][10],
          mul [10][10];
     int r1, c1, r2, c2, i,j, k;
    printf (" Enter rows and colums for first
           matrix: ");
    Scanf ("%d %d ", &r1, &c1);
   printf ("Enter rowsand colums for second matrix:");
   Scanf ("%d %d", &r2, &c2);
   printf ("\n Enter elements of first matrix: \n");
   for (i=0; i<r1 ; i++)
     for (j=0; j<c1 ; j++)
        scanf ("%d", &a [i][j]);
   printf ("\n Enter elements of second matrix:\n");
  for (i=0; i<r2 ; i++)
     for (j=0;  j<c2 ; j++)
       Scanf  ("%d" , &a [i] [j] );

~~printf ("Er~~
      if (r1 == r2 && c1 == c2){
     printf ("\n Addition of matrices: \n");
     for (i=0; i<r1 ; i++) {
         for (j=0 ; j < c1 ; j++) {
         sum [i] [j] = a [i] [j] + b[i] [j];
         printf ("%d\t", sum [i][j]);
      }
       printf ("\n");
      }
   } else {
    printf ("\n Addition not possible (different dimensions)
                                      \n");
    }
     if (r1 == r2 && c1 == c2) {
       printf ("\n Subtraction of matrices: \n");
     for (i = 0; i< r1 ; i++) {
        for (j = 0; j< c1; j++) {
          sub [i] [j] =   a[i] [j] - b [i] [j];
        printf ("%d \t", sub [i] [j]);
      }
        printf ("\n");
      }
   } else {
    prinf ("\n Subtraction not possible (different dimension)\n
   }
    if (c1 == r2) {
     printf ("\nMultiplication of matrices : \n");
for (i = 0; i < r1; i++) {
    for (j = 0; j < c2; j++) {
        mul [i] [j] = 0;
        for (k = 0; k < c1; k++)
            mul [i][j] += a [i] [k] * b [k] [j];
        printf ("%d \t", mul [i] [j]);
    }
    printf ("\n");
  }
} else {
    printf ("\nMultiplication not possible (column of
          A ≠ row of B ) \n");
  }
  return 0;
}
```

⇒ Output :-

```
Enter rows and columns for first matrix : 2, 3
Enter rows and columns for second matrix : 2, 2
Enter elements of first matrix:  1  4  7
                                 3  6  9

Enter elements of second matrix:  5  6
                                  8  9


Addition not possible (different dimensions)
Subtraction not possible (different dimensions)
multiplication not possible (column of A ≠ row of B)
```

8.
```
#include<stdio.h>
 void print 1 ToN (int i, int n) {
   if (i>n) return;
   printf ("%d", i);
   print1ToN (i+1, n);
 }
  void printNTo1 (int n){
   if (n == 0) return;
   printf ("%d", n);
   printNTo1 (n-1);
  }
  int  sum (int n) {
    if (n==0) return 0;
    return n+sum (n-1);
  }
  int fact (int n) {
     if (n==0) return 1;
    return n* fact (n-1);
  }
  void table (int n, int i) {
      if (i>10) return;
      printf ("%d x %d = %d\n", n, i, n*i);
     table (n, i+1);
   }
   int fib (int n) {
     if (n <= 1) return n;
     return fib (n-1) + fib (n-2);
  }
   int gcd (int a, int b) {
     if (b==0) return a;
    return gcd (b, a % b);
  }
  int  countDigits (int n) {
```

```
  if (n==0) return 0;
  return 1+ CountDigits (n/10);
}
intmain () {
   int n , a , b, Choice ;
   do {
     printf (" \n\n ==== menu (Recursion Programs)
                 ====\n");
    printf ("1. Print 1 to N\n");
    printf ("2. Print N to 1 \n");
    printf ("3. sum of first N natural numbers \n ");
    printf (" 4. fuctorial \n");
    printf ("5. Multiplication Table \n");
    printf ("6. fibonacci series \n");
    printf ("7. GCD of two numbers \n");
    printf ("8. Count digits \n");
    printf ("9. Exit \n");
     printf ("Enter your choice : ");
    scanf ("%d ", &choice);
~~case~~ switch (choice) {
 Case 1 :
     printf ("Enter N: ");
     Scanf ("%d", &n);
    print1ToN (1,n);
     break;
Case 2 :
     printf ("Enter N:");
    Sconf ("%d", &n);
     printNTo1 (n);
     break;
Case 3;
    Printf (" Enter N:");
    Scanf ("%d", &n);
    Printf ("Sum = %.d", sum(n)); break;
case 4:
    printf ("Enter number: ");
    scanf ("%d", &n);
    printf ("Factorial = %d", fact (n));
    break;
Case 5;
    printf ("Enter number: ");
    Scanf ("%d", &n);
    table (n,1);
    break;
Case 6:
    printf ("Enter N: ");
    scanf ("%d", &n);
    for (int i = 0; i<n ; i++)
        printf ("%d", fib(i));
    break;
Case 7:
    printf ("Enter two numbers: ");
    scanf ("%d %d", &a, &b);
    printf ("GCD = %d", gcd (a,b));
    break;
Case 8:
  printf ("Enter a number: ");
  Scanf ("%d", &n);
  printf (" Digits = %d", contDigits (n));
  break;
Case 9:
    printf ("Exiting ...\n");
    break;
 default :
    printf ("Invalid choice!");
}
} while (choice != 9);
  return 0;    }
```

⇒ Output:

```
1. Print 1 to N
2 print N to 1
3. sum of first N natural numbers
4. Factorial
5. Multiplication Table
6. Fibonacci series
7 GCD of two numbers
8. Count digits
9. Exit.

Enter your Choice : 3
Enter N : 5
sum = 15
```