Assignment 12 and 13

Name: Om Vivek Gharge
PRN: 2020BTEIT00041
Branch: Information Technology

Q1. Write a program in C to find the sum of the series 1!/1+2!/2+3!/3+4!/4+5!/5 using the function.

```c
#include <stdio.h>

int fact(int);
void main()
{
    int sum;
    sum=fact(1)/1+fact(2)/2+fact(3)/3+fact(4)/4+fact(5)/5;
    printf("\n\n Function : find the sum of 1!/1+2!/2+3!/3+4!/4+5!/5 :\n");
    printf("-------------------------------------------------------------\n");
    printf("The sum of the series is : %d\n\n",sum);
}

int fact(int n)
    {
        int num=0,f=1;
        while(num<=n-1)
        {
            f =f+f*num;
            num++;
        }
    return f;
    }
```

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL                    2: Code

Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Users\omgha> cd "C:\Users\omgha\AppData\Local\Temp\" ; if ($?) { gcc tempCodeRunnerFile.c -o tempCodeRunnerFile } ; if ($?) { .\tempCodeRunnerFile }


 Function : find the sum of 1!/1+2!/2+3!/3+4!/4+5!/5 :
-------------------------------------------------------------
The sum of the series is : 34

PS C:\Users\omgha\AppData\Local\Temp>
```

Q2. Write a program in C to find the square of any number using the function.

```c
#include <stdio.h>

double square(double num)
{
    return (num * num);
}
int main()
{
    int num;
    double n;
    printf("\n\n Function : find square of any number :\n");
    printf("-------------------------------------------------\n");

    printf("Input any number for square : ");
    scanf("%d", &num);
    n = square(num);
    printf("The square of %d is : %.2f\n", num, n);
    return 0;
}
```

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL                                  2: Code

Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Users\omgha> cd "C:\Users\omgha\AppData\Local\Temp\" ; if ($?) { gcc tempCodeRunnerFile.c -o tempCodeRunnerFile } ; if ($?) { .\tempCodeRunnerFile }


 Function : find square of any number :
-------------------------------------------------
Input any number for square : 25
The square of 25 is : 625.00
PS C:\Users\omgha\AppData\Local\Temp>
```

Q3. Write a program in C to swap two numbers using function.

```c
#include<stdio.h>

void swap(int *,int *);
int main()
{

    int n1,n2;
    printf("\n\n Function : swap two numbers using function :\n");
    printf("-------------------------------------------------\n");
    printf("Input 1st number : ");
    scanf("%d",&n1);
    printf("Input 2nd number : ");
    scanf("%d",&n2);

    printf("Before swapping: n1 = %d, n2 = %d ",n1,n2);
    //pass the address of both variables to the function.
    swap(&n1,&n2);

    printf("\nAfter swapping: n1 = %d, n2 = %d \n\n",n1,n2);
    return 0;
}

void swap(int *p,int *q)
{
    //p=&n1 so p store the address of n1, so *p store the value of n1
    //q=&n2 so q store the address of n2, so *q store the value of n2

    int tmp;
    tmp = *p; // tmp store the value of n1
    *p=*q;    // *p store the value of *q that is value of n2
    *q=tmp;   // *q store the value of tmp that is the value of n1
}
```

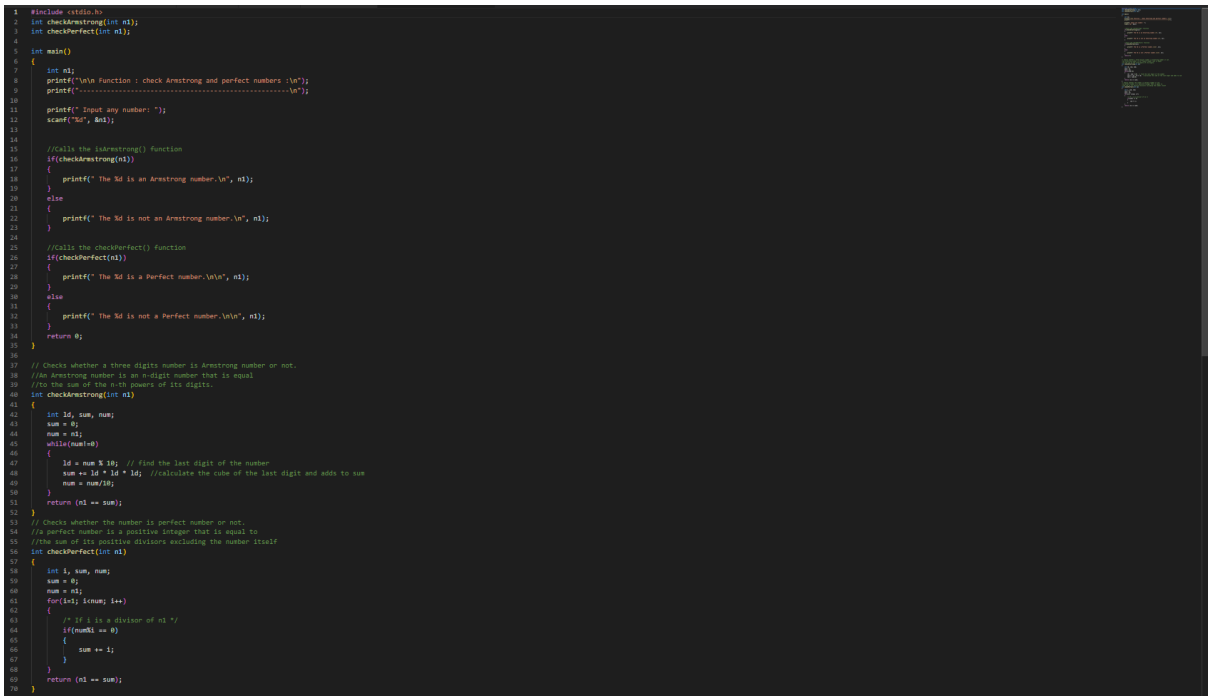
```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Users\omgha> cd "C:\Users\omgha\AppData\Local\Temp\" ; if ($?) { gcc tempCodeRunnerFile.c -o tempCodeRunnerFile } ; if ($?) { .\tempCodeRunnerFile }


 Function : swap two numbers using function :
-------------------------------------------------
Input 1st number : 5
Input 2nd number : 6
Before swapping: n1 = 5, n2 = 6
After swapping: n1 = 6, n2 = 5
```

Q4. Write a program in C to get the largest element of an array using the function.

```c
#include<stdio.h>
#define MAX 100

int findMaxElem(int []);
int n;

int main()
{
    int arr1[MAX],mxelem,i;
    printf("\n\n Function : get largest element of an array :\n");
    printf("-------------------------------------------------\n");

       printf(" Input the number of elements to be stored in the array :");
       scanf("%d",&n);

       printf(" Input %d elements in the array :\n",n);
       for(i=0;i<n;i++)
        {
          printf(" element - %d : ",i);
          scanf("%d",&arr1[i]);
        }
    mxelem=findMaxElem(arr1);

    printf(" The largest element in the array is : %d\n\n",mxelem);
    return 0;
}
int findMaxElem(int arr1[])
{
    int i=1,mxelem;
    mxelem=arr1[0];
    while(i < n)
    {
      if(mxelem<arr1[i])
           mxelem=arr1[i];
      i++;
    }
    return mxelem;
}
```

```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Users\omgha> cd "C:\Users\omgha\AppData\Local\Temp\" ; if ($?) { gcc tempCodeRunnerFile.c -o tempCodeRunnerFile } ; if ($?) { .\tempCodeRunnerFile }


 Function : get largest element of an array :
-------------------------------------------------
 Input the number of elements to be stored in the array :5
 Input 5 elements in the array :
 element - 0 : 1
 element - 1 : 2
 element - 2 : 3
 element - 3 : 4
 element - 4 : 5
 The largest element in the array is : 5
```

Q5. Write a program in C to check armstrong and perfect numbers using the function.



```c
#include <stdio.h>
int checkArmstrong(int n1);
int checkPerfect(int n1);

int main()
{
    int n1;
    printf("\n\n Function : check Armstrong and perfect numbers :\n");
    printf("-------------------------------------------------------\n");

    printf(" Input any number: ");
    scanf("%d", &n1);


    //Calls the isArmstrong() function
    if(checkArmstrong(n1))
    {
        printf(" The %d is an Armstrong number.\n", n1);
    }
    else
    {
        printf(" The %d is not an Armstrong number.\n", n1);
    }

    //Calls the checkPerfect() function
    if(checkPerfect(n1))
    {
        printf(" The %d is a Perfect number.\n\n", n1);
    }
    else
    {
        printf(" The %d is not a Perfect number.\n\n", n1);
    }
    return 0;
}

// Checks whether a three digits number is Armstrong number or not.
//An Armstrong number is an n-digit number that is equal
//to the sum of the n-th powers of its digits.
int checkArmstrong(int n1)
{
    int ld, sum, num;
    sum = 0;
    num = n1;
    while(num!=0)
    {
        ld = num % 10;  // find the last digit of the number
        sum += ld * ld * ld; //calculate the cube of the last digit and adds to sum
        num = num/10;
    }
    return (n1 == sum);
}
// Checks whether the number is perfect number or not.
//a perfect number is a positive integer that is equal to
//the sum of its positive divisors excluding the number itself
int checkPerfect(int n1)
{
    int i, sum, num;
    sum = 0;
    num = n1;
    for(i=1; i<num; i++)
    {
        /* if i is a divisor of n1 */
        if(num%i == 0)
        {
            sum += i;
        }
    }
    return (n1 == sum);
}
```

```
PS C:\Users\omgha> cd "C:\Users\omgha\AppData\Local\Temp\" ; if ($?) { gcc tempCodeRunnerFile.c -o tempCodeRunnerFile } ; if ($?) { .\tempCodeRunnerFile }


 Function : check Armstrong and perfect numbers :
-------------------------------------------------------
 Input any number: 153
 The 153 is an Armstrong number.
 The 153 is not a Perfect number.

PS C:\Users\omgha\AppData\Local\Temp> cd "C:\Users\omgha\AppData\Local\Temp\" ; if ($?) { gcc tempCodeRunnerFile.c -o tempCodeRunn
erfile }


 Function : check Armstrong and perfect numbers :
-------------------------------------------------------
 Input any number: 6
 The 6 is not an Armstrong number.
 The 6 is a Perfect number.
```

Q6. Write a C programming to find out maximum and minimum of some values using function which will return an array.

```c
# include <stdio.h>
# define max 10
int *maxmin(int ar[], int v);
int main()
{
    int  arr[max];
    int n,i, *p;
    printf("Number of values you want to input: ");
    scanf("%d",&n);
    printf("Input %d values\n", n);
    for(i=0;i<n;i++)
        scanf("%d",&arr[i]);
    p=maxmin(arr,n);
    printf("Minimum value is: %d\n",*p++);
    printf("Maximum value is: %d\n",*p);
}
int *maxmin(int arra1[], int v)
{
    int i;
    static int result_mm[2];
    result_mm[0]=arra1[0];
    result_mm[1]=arra1[0];
    for (i=1;i<v;i++)
    {
        if(result_mm[0] > arra1[i])
            result_mm[0]=arra1[i];
        if(result_mm[1]< arra1[i])
            result_mm[1]= arra1[i];
    }
    return result_mm;
}
```

```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Users\omgha> cd "C:\Users\omgha\AppData\Local\Temp\" ; if ($?) { gcc tempCodeRunnerFile.c -o tempCodeRunnerFile } ; if ($?) { .\tempCodeRunnerFile }
Number of values you want to input: 5
Input 5 values
65
33
25
11
0
Minimum value is: 0
Maximum value is: 65
PS C:\Users\omgha\AppData\Local\Temp>
```

Q7. Write a program in C to print first 50 natural numbers using recursion.

```c
#include<stdio.h>
int  numPrint(int);
int main()
{
    int n = 1;
    printf("\n\n Recursion : print first 50 natural numbers :\n");
    printf("-------------------------------------------------\n");
    printf(" The natural numbers are :");
    numPrint(n);
    printf("\n\n");
    return 0;
}
int numPrint(int n)
{
    if(n<=50)
    {
        printf(" %d ",n);
        numPrint(n+1);
    }
}
```

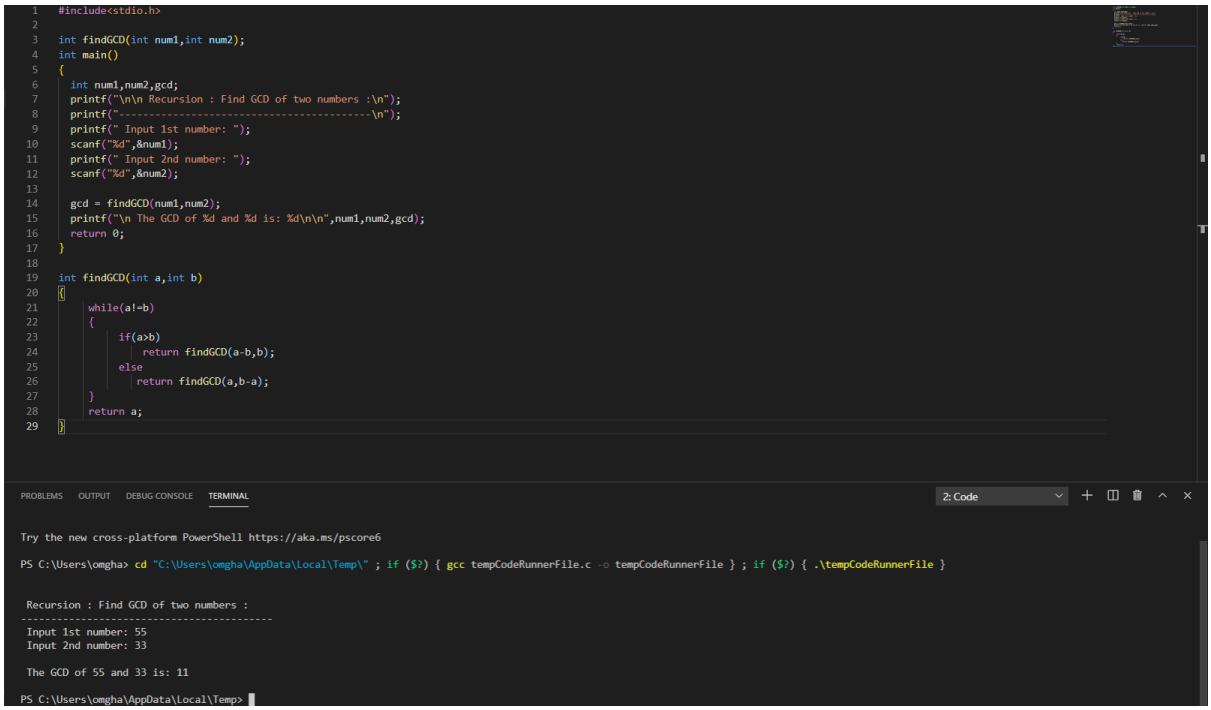
```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL                                  2: Code

Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Users\omgha> cd "C:\Users\omgha\AppData\Local\Temp\" ; if ($?) { gcc tempCodeRunnerFile.c -o tempCodeRunnerFile } ; if ($?) { .\tempCodeRunnerFile }


 Recursion : print first 50 natural numbers :
-------------------------------------------------
 The natural numbers are : 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  31  32  33  34  35  36  37
 38  39  40  41  42  43  44  45  46  47  48  49  50

PS C:\Users\omgha\AppData\Local\Temp>
```

Q8. Write a program in C to print the array elements using recursion.

```c
#include <stdio.h>
#define MAX 100

void ArrayElement(int arr1[], int st, int l);

int main()
{
    int arr1[MAX];
    int n, i;
    printf("\n\n Recursion : Print the array elements :\n");
    printf("-----------------------------------------\n");

    printf(" Input the number of elements to be stored in the array :");
    scanf("%d",&n);

    printf(" Input %d elements in the array :\n",n);
    for(i=0;i<n;i++)
      {
        printf(" element - %d : ",i);
        scanf("%d",&arr1[i]);
      }

    printf(" The elements in the array are : ");
    ArrayElement(arr1, 0, n);//call the function ArrayElement
    printf("\n\n");
    return 0;
}

void ArrayElement(int arr1[], int st, int l)
{
    if(st >= l)
        return;

    //Prints the current array element
    printf("%d  ", arr1[st]);

    /* Recursively call ArrayElement to print next element in the array */
    ArrayElement(arr1, st+1, l);//calling the function  ArrayElement itself
}
```

```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Users\omgha> cd "C:\Users\omgha\AppData\Local\Temp\" ; if ($?) { gcc tempCodeRunnerFile.c -o tempCodeRunnerFile } ; if ($?) { .\tempCodeRunnerFile }


 Recursion : Print the array elements :
-----------------------------------------
 Input the number of elements to be stored in the array :5
 Input 5 elements in the array :
 element - 0 : 55
 element - 1 : 88
 element - 2 : 64
 element - 3 : 22
 element - 4 : 33
 The elements in the array are : 55  88  64  22  33

PS C:\Users\omgha\AppData\Local\Temp>
```

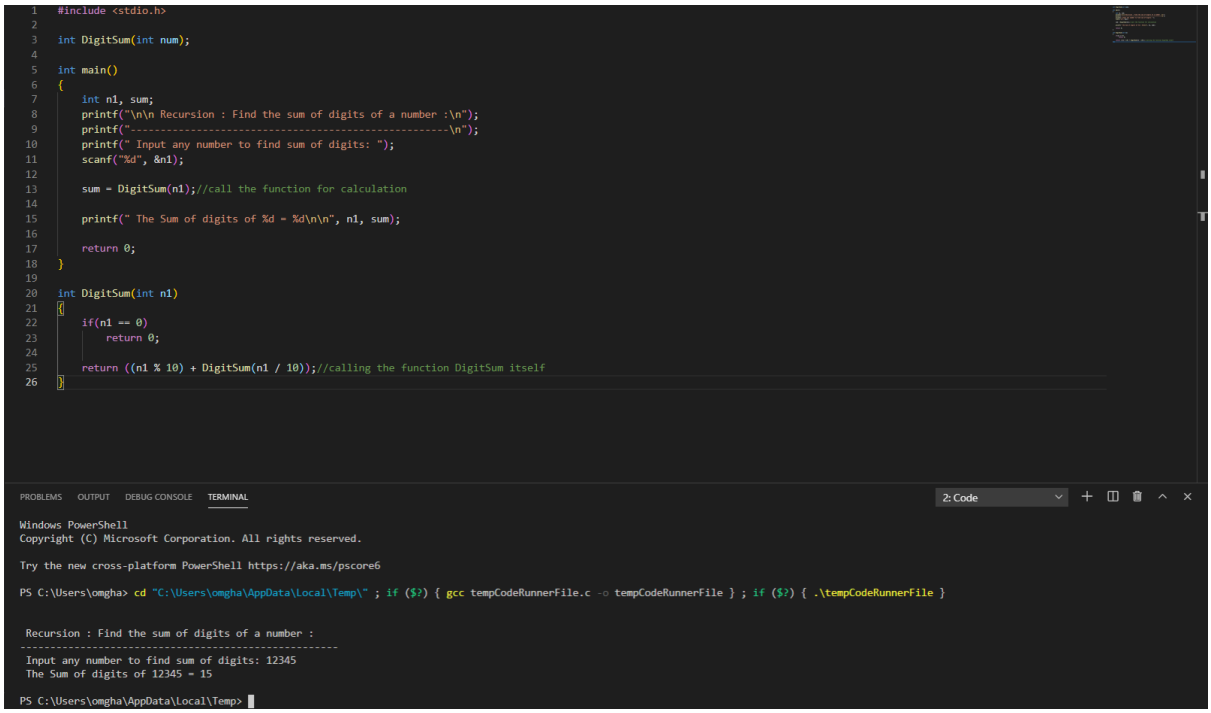
Q9. Write a program in C to find GCD of two numbers using recursion.



```c
#include<stdio.h>

int findGCD(int num1,int num2);
int main()
{
  int num1,num2,gcd;
  printf("\n\n Recursion : Find GCD of two numbers :\n");
  printf("-----------------------------------------\n");
  printf(" Input 1st number: ");
  scanf("%d",&num1);
  printf(" Input 2nd number: ");
  scanf("%d",&num2);

  gcd = findGCD(num1,num2);
  printf("\n The GCD of %d and %d is: %d\n\n",num1,num2,gcd);
  return 0;
}

int findGCD(int a,int b)
{
    while(a!=b)
    {
        if(a>b)
            return findGCD(a-b,b);
        else
            return findGCD(a,b-a);
    }
    return a;
}
```

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL                    2: Code

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Users\omgha> cd "C:\Users\omgha\AppData\Local\Temp\" ; if ($?) { gcc tempCodeRunnerFile.c -o tempCodeRunnerFile } ; if ($?) { .\tempCodeRunnerFile }


 Recursion : Find GCD of two numbers :
-----------------------------------------
 Input 1st number: 55
 Input 2nd number: 33

 The GCD of 55 and 33 is: 11

PS C:\Users\omgha\AppData\Local\Temp>
```

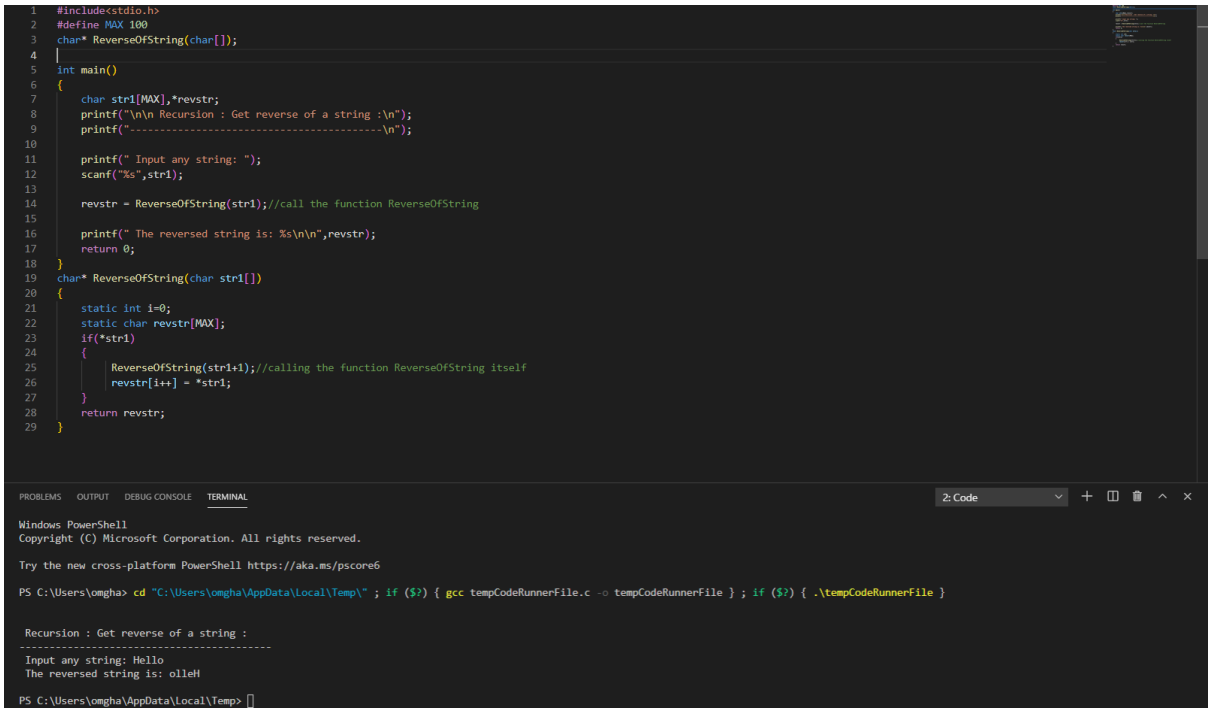
Q10. Write a program in C to find the sum of digits of a number using recursion.



```c
#include <stdio.h>

int DigitSum(int num);

int main()
{
    int n1, sum;
    printf("\n\n Recursion : Find the sum of digits of a number :\n");
    printf("-------------------------------------------------------\n");
    printf(" Input any number to find sum of digits: ");
    scanf("%d", &n1);

    sum = DigitSum(n1);//call the function for calculation

    printf(" The Sum of digits of %d = %d\n\n", n1, sum);

    return 0;
}

int DigitSum(int n1)
{
    if(n1 == 0)
        return 0;

    return ((n1 % 10) + DigitSum(n1 / 10));//calling the function DigitSum itself
}
```

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL                                  2: Code

Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Users\omgha> cd "C:\Users\omgha\AppData\Local\Temp\" ; if ($?) { gcc tempCodeRunnerFile.c -o tempCodeRunnerFile } ; if ($?) { .\tempCodeRunnerFile }


 Recursion : Find the sum of digits of a number :
-------------------------------------------------------
 Input any number to find sum of digits: 12345
 The Sum of digits of 12345 = 15

PS C:\Users\omgha\AppData\Local\Temp>
```

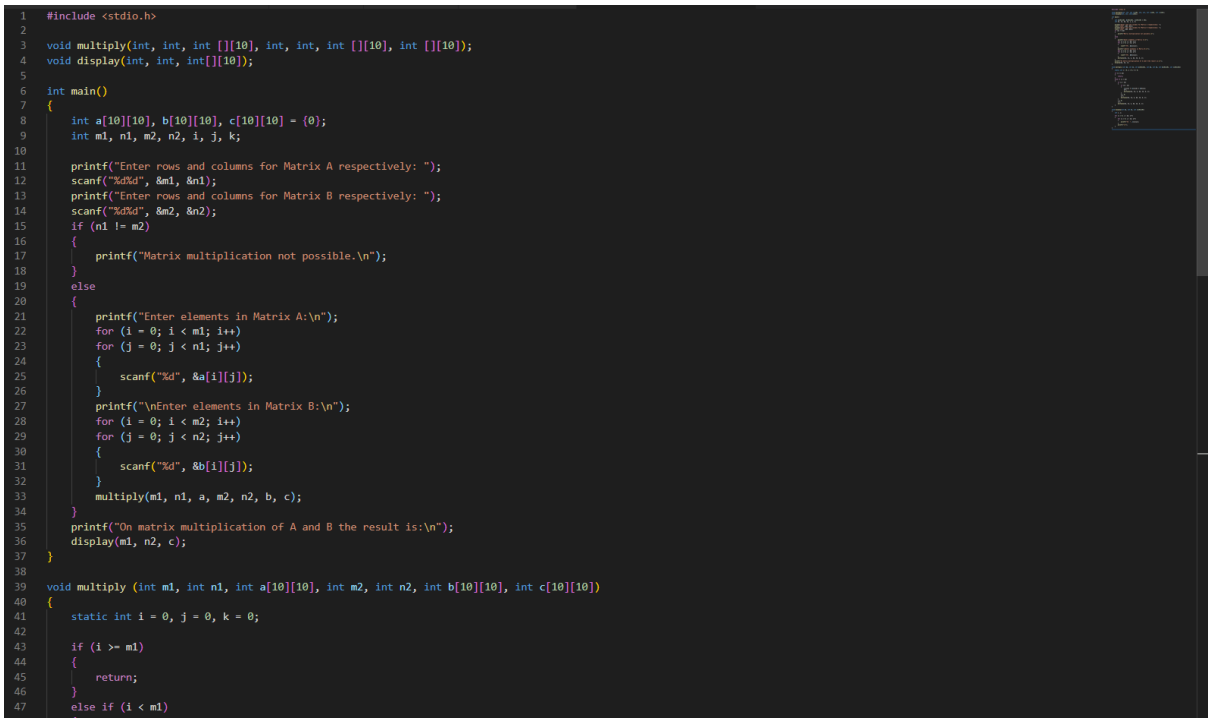
Q.11 Write a program in C to reverse a string using recursion**.**



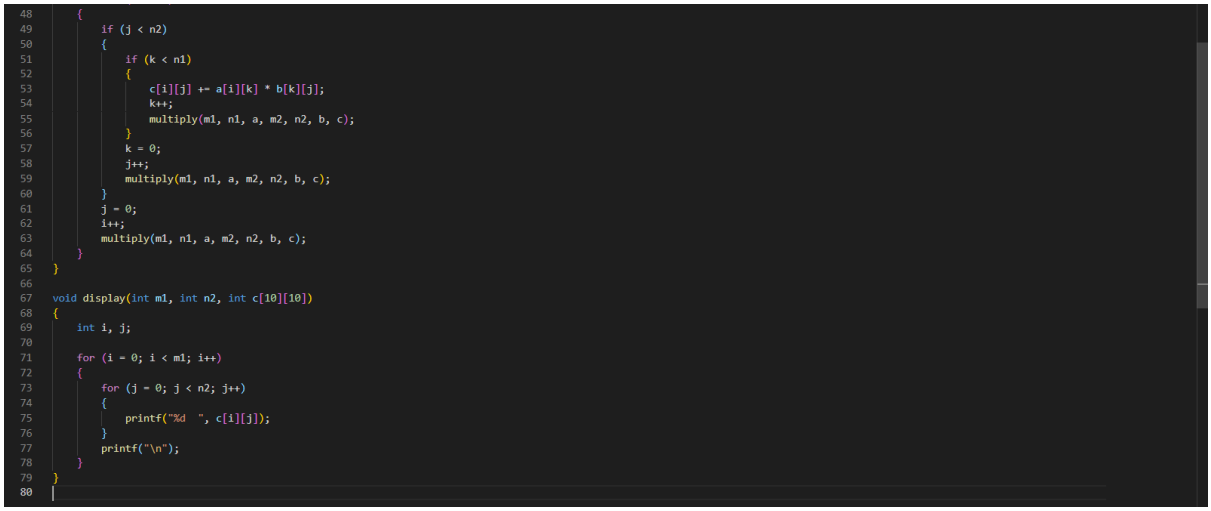
```c
#include<stdio.h>
#define MAX 100
char* ReverseOfString(char[]);

int main()
{
    char str1[MAX],*revstr;
    printf("\n\n Recursion : Get reverse of a string :\n");
    printf("-----------------------------------------\n");

    printf(" Input any string: ");
    scanf("%s",str1);

    revstr = ReverseOfString(str1);//call the function ReverseOfString

    printf(" The reversed string is: %s\n\n",revstr);
    return 0;
}
char* ReverseOfString(char str1[])
{
    static int i=0;
    static char revstr[MAX];
    if(*str1)
    {
        ReverseOfString(str1+1);//calling the function ReverseOfString itself
        revstr[i++] = *str1;
    }
    return revstr;
}
```

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL                                  2: Code

Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Users\omgha> cd "C:\Users\omgha\AppData\Local\Temp\" ; if ($?) { gcc tempCodeRunnerFile.c -o tempCodeRunnerFile } ; if ($?) { .\tempCodeRunnerFile }


 Recursion : Get reverse of a string :
-----------------------------------------
 Input any string: Hello
 The reversed string is: olleH

PS C:\Users\omgha\AppData\Local\Temp>
```

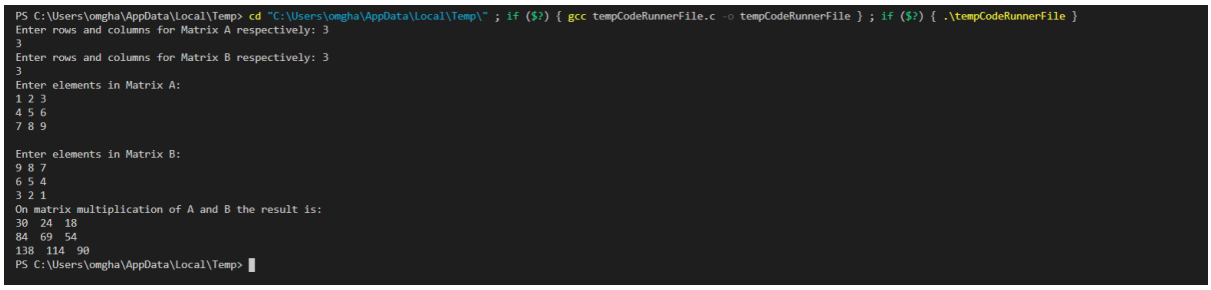
Q.12 Write a program in C to multiply two matrix using recursion.



```c
#include <stdio.h>

void multiply(int, int, int [][10], int, int, int [][10], int [][10]);
void display(int, int, int[][10]);

int main()
{
    int a[10][10], b[10][10], c[10][10] = {0};
    int m1, n1, m2, n2, i, j, k;

    printf("Enter rows and columns for Matrix A respectively: ");
    scanf("%d%d", &m1, &n1);
    printf("Enter rows and columns for Matrix B respectively: ");
    scanf("%d%d", &m2, &n2);
    if (n1 != m2)
    {
        printf("Matrix multiplication not possible.\n");
    }
    else
    {
        printf("Enter elements in Matrix A:\n");
        for (i = 0; i < m1; i++)
        for (j = 0; j < n1; j++)
        {
            scanf("%d", &a[i][j]);
        }
        printf("\nEnter elements in Matrix B:\n");
        for (i = 0; i < m2; i++)
        for (j = 0; j < n2; j++)
        {
            scanf("%d", &b[i][j]);
        }
        multiply(m1, n1, a, m2, n2, b, c);
    }
    printf("On matrix multiplication of A and B the result is:\n");
    display(m1, n2, c);
}

void multiply (int m1, int n1, int a[10][10], int m2, int n2, int b[10][10], int c[10][10])
{
    static int i = 0, j = 0, k = 0;

    if (i >= m1)
    {
        return;
    }
    else if (i < m1)
```



```c
    {
        if (j < n2)
        {
            if (k < n1)
            {
                c[i][j] += a[i][k] * b[k][j];
                k++;
                multiply(m1, n1, a, m2, n2, b, c);
            }
            k = 0;
            j++;
            multiply(m1, n1, a, m2, n2, b, c);
        }
        j = 0;
        i++;
        multiply(m1, n1, a, m2, n2, b, c);
    }
}

void display(int m1, int n2, int c[10][10])
{
    int i, j;

    for (i = 0; i < m1; i++)
    {
        for (j = 0; j < n2; j++)
        {
            printf("%d  ", c[i][j]);
        }
        printf("\n");
    }
}
```



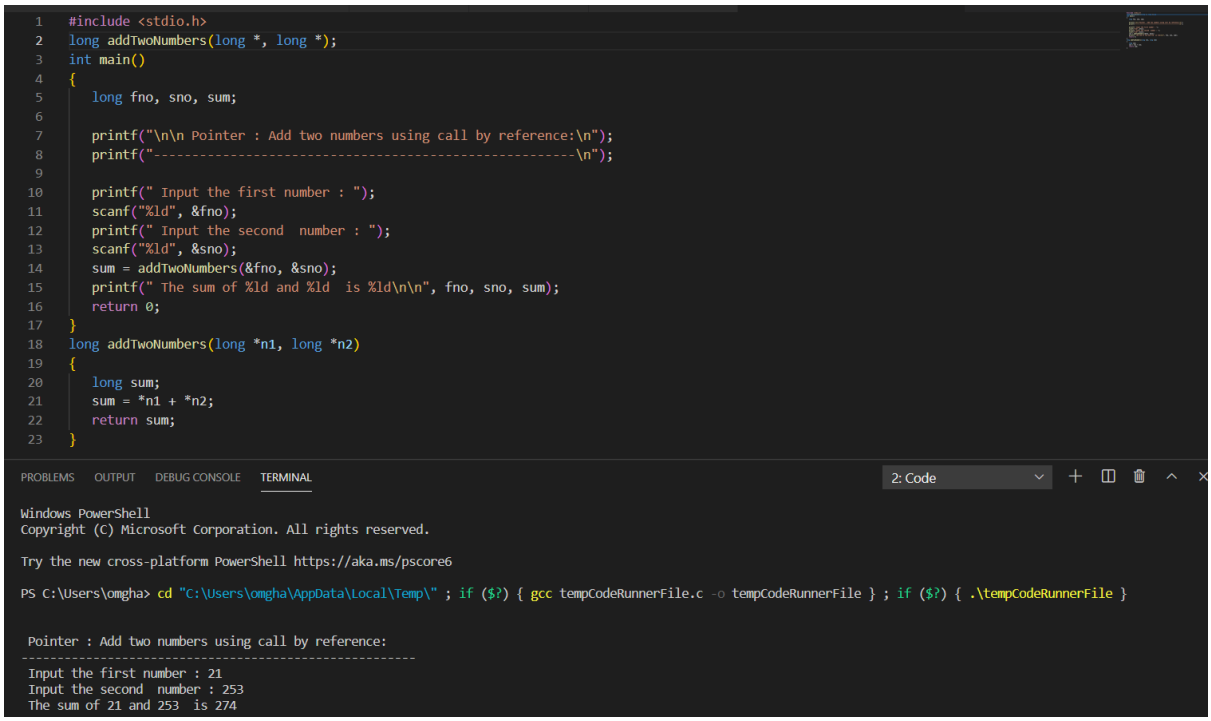
```
PS C:\Users\omgha\AppData\Local\Temp> cd "C:\Users\omgha\AppData\Local\Temp\" ; if ($?) { gcc tempCodeRunnerFile.c -o tempCodeRunnerFile } ; if ($?) { .\tempCodeRunnerFile }
Enter rows and columns for Matrix A respectively: 3
3
Enter rows and columns for Matrix B respectively: 3
3
Enter elements in Matrix A:
1 2 3
4 5 6
7 8 9

Enter elements in Matrix B:
9 8 7
6 5 4
3 2 1
On matrix multiplication of A and B the result is:
30  24  18
84  69  54
138  114  90
PS C:\Users\omgha\AppData\Local\Temp>
```

Q.13 Write a program in C to show the basic declaration of pointer.

```c
#include <stdio.h>
void main(void)
{
int m=10,n,o;
int *z=&m ;

    printf("\n\n Pointer : Show the basic declaration of pointer :\n");
    printf("-------------------------------------------------------\n");
    printf(" Here is m=10, n and o are two integer variable and *z is an integer");
    printf("\n\n z stores the address of m  = %p\n",  z); // z is a pointer so %p would print the address
    printf("\n *z stores the value of m = %i\n",   *z);
    printf("\n &m is the address of m = %p\n",   &m); // &m gives the address of the integer variable m
                        // so %p is the specifier for that address
    printf("\n &n stores the address of n = %p\n",   &n);
    printf("\n &o  stores the address of o = %p\n",   &o);
    printf("\n &z stores the address of z = %p\n\n", &z); // &z gives the address, where the pointer z is
                        // stored -> still an address -> %p is the right
                        // specifier
}
```

```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Users\omgha> cd "C:\Users\omgha\AppData\Local\Temp\" ; if ($?) { gcc tempCodeRunnerFile.c -o tempCodeRunnerFile } ; if ($?) { .\tempCodeRunnerFile }


 Pointer : Show the basic declaration of pointer :
-------------------------------------------------------
 Here is m=10, n and o are two integer variable and *z is an integer

 z stores the address of m  = 000000000061FE1C

 *z stores the value of m = 10

 &m is the address of m = 000000000061FE1C

 &n stores the address of n = 000000000061FE18

 &o  stores the address of o = 000000000061FE14

 &z stores the address of z = 000000000061FE08

PS C:\Users\omgha\AppData\Local\Temp>
```

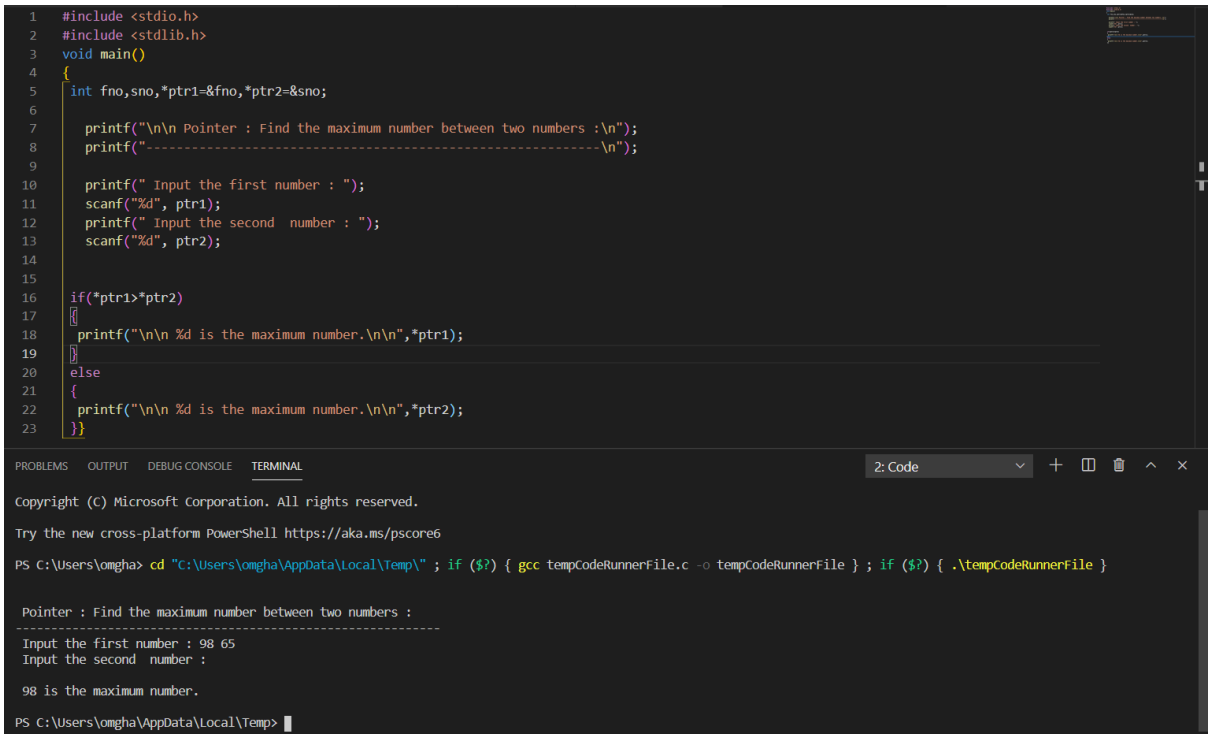
Q.14 Write a program in C to add numbers using call by reference.



```c
#include <stdio.h>
long addTwoNumbers(long *, long *);
int main()
{
   long fno, sno, sum;

   printf("\n\n Pointer : Add two numbers using call by reference:\n");
   printf("-------------------------------------------------------\n");

   printf(" Input the first number : ");
   scanf("%ld", &fno);
   printf(" Input the second  number : ");
   scanf("%ld", &sno);
   sum = addTwoNumbers(&fno, &sno);
   printf(" The sum of %ld and %ld  is %ld\n\n", fno, sno, sum);
   return 0;
}
long addTwoNumbers(long *n1, long *n2)
{
   long sum;
   sum = *n1 + *n2;
   return sum;
}
```

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL                    2: Code

Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Users\omgha> cd "C:\Users\omgha\AppData\Local\Temp\" ; if ($?) { gcc tempCodeRunnerFile.c -o tempCodeRunnerFile } ; if ($?) { .\tempCodeRunnerFile }


 Pointer : Add two numbers using call by reference:
-------------------------------------------------------
 Input the first number : 21
 Input the second  number : 253
 The sum of 21 and 253  is 274
```

Q.15 Write a program in C to find the maximum number between two numbers using a pointer.



```c
#include <stdio.h>
#include <stdlib.h>
void main()
{
 int fno,sno,*ptr1=&fno,*ptr2=&sno;

   printf("\n\n Pointer : Find the maximum number between two numbers :\n");
   printf("-------------------------------------------------------------\n");

   printf(" Input the first number : ");
   scanf("%d", ptr1);
   printf(" Input the second  number : ");
   scanf("%d", ptr2);


 if(*ptr1>*ptr2)
 {
  printf("\n\n %d is the maximum number.\n\n",*ptr1);
 }
 else
 {
  printf("\n\n %d is the maximum number.\n\n",*ptr2);
 }}
```

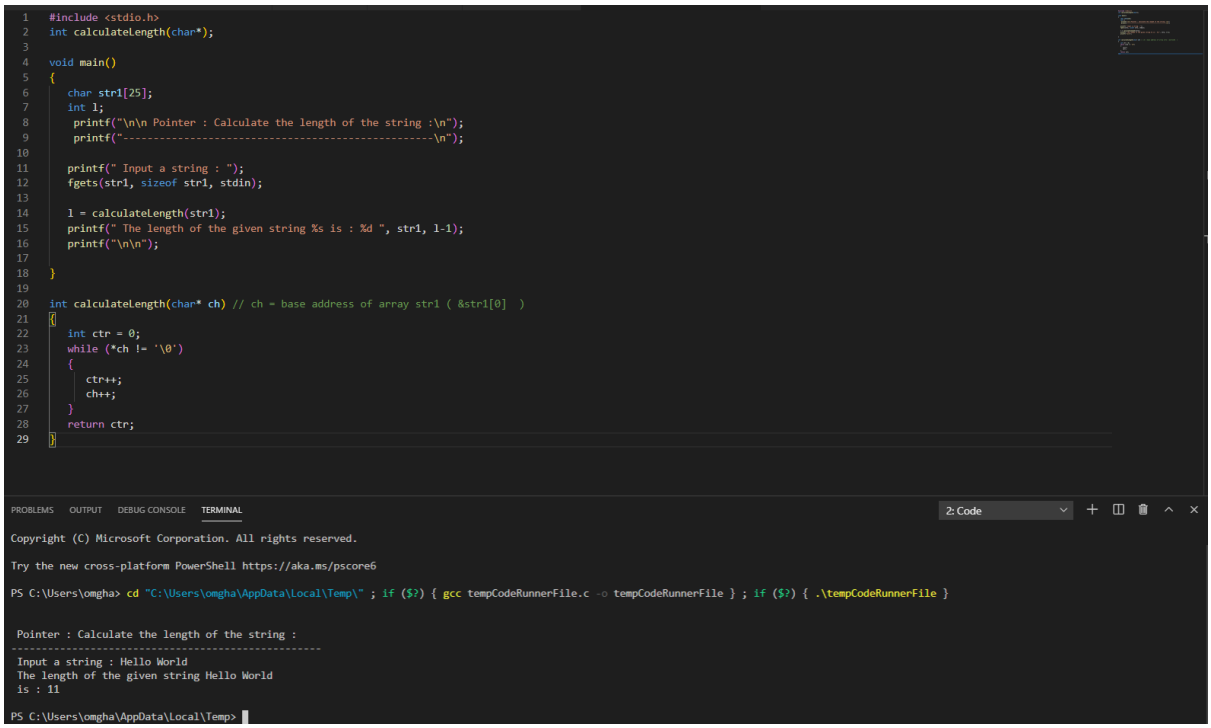
```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL                                  2: Code

Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Users\omgha> cd "C:\Users\omgha\AppData\Local\Temp\" ; if ($?) { gcc tempCodeRunnerFile.c -o tempCodeRunnerFile } ; if ($?) { .\tempCodeRunnerFile }


 Pointer : Find the maximum number between two numbers :
-------------------------------------------------------------
 Input the first number : 98 65
 Input the second  number :

 98 is the maximum number.

PS C:\Users\omgha\AppData\Local\Temp>
```

Q.16 Write a program in C to print all permutations of a given string using pointers.

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

void changePosition(char *ch1, char *ch2)
{
    char tmp;
    tmp = *ch1;
    *ch1 = *ch2;
    *ch2 = tmp;
}
void charPermu(char *cht, int stno, int endno)
{
   int i;
   if (stno == endno)
    printf("%s  ", cht);
   else
   {
       for (i = stno; i <= endno; i++)
       {
          changePosition((cht+stno), (cht+i));
          charPermu(cht, stno+1, endno);
          changePosition((cht+stno), (cht+i));
       }
   }
}

int main()
{
    char str[] = "abcd";
   printf("\n\n Pointer : Generate permutations of a given string :\n");
   printf("------------------------------------------------------------\n");
    int n = strlen(str);
    printf(" The permutations of the string are : \n");
    charPermu(str, 0, n-1);
    printf("\n\n");
    return 0;
}
```

```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Users\omgha> cd "C:\Users\omgha\AppData\Local\Temp\" ; if ($?) { gcc tempCodeRunnerFile.c -o tempCodeRunnerFile } ; if ($?) { .\tempCodeRunnerFile }


 Pointer : Generate permutations of a given string :
------------------------------------------------------------
 The permutations of the string are :
abcd  abdc  acbd  acdb  adcb  adbc  bacd  badc  bcad  bcda  bdca  bdac  cbad  cbda  cabd  cadb  cdab  cdba  dbca  dbac  dcba  dcab  dacb  dabc

PS C:\Users\omgha\AppData\Local\Temp>
```

Q.17 Write a program in C to Calculate the length of the string using a pointer.



```c
#include <stdio.h>
int calculateLength(char*);

void main()
{
   char str1[25];
   int l;
    printf("\n\n Pointer : Calculate the length of the string :\n");
    printf("-------------------------------------------------\n");

   printf(" Input a string : ");
   fgets(str1, sizeof str1, stdin);

   l = calculateLength(str1);
   printf(" The length of the given string %s is : %d ", str1, l-1);
   printf("\n\n");

}

int calculateLength(char* ch) // ch = base address of array str1 ( &str1[0]  )
{
   int ctr = 0;
   while (*ch != '\0')
   {
      ctr++;
      ch++;
   }
   return ctr;
}
```

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL                                  2: Code

Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Users\omgha> cd "C:\Users\omgha\AppData\Local\Temp\" ; if ($?) { gcc tempCodeRunnerFile.c -o tempCodeRunnerFile } ; if ($?) { .\tempCodeRunnerFile }


 Pointer : Calculate the length of the string :
-------------------------------------------------
 Input a string : Hello World
 The length of the given string Hello World
 is : 11

PS C:\Users\omgha\AppData\Local\Temp>
```

Q.18 Write a program in C to find the factorial of a given number using pointers.

```c
#include <stdio.h>
void findFact(int,int*);
int main()
{
        int fact;
        int num1;
       printf("\n\n Pointer : Find the factorial of a given number :\n");
       printf("-----------------------------------------------------\n");
       printf(" Input a number : ");
       scanf("%d",&num1);

        findFact(num1,&fact);
        printf(" The Factorial of %d is : %d \n\n",num1,fact);
        return 0;
       }

void findFact(int n,int *f)
       {
       int i;

      *f =1;
      for(i=1;i<=n;i++)
      *f=*f*i;
      }
```

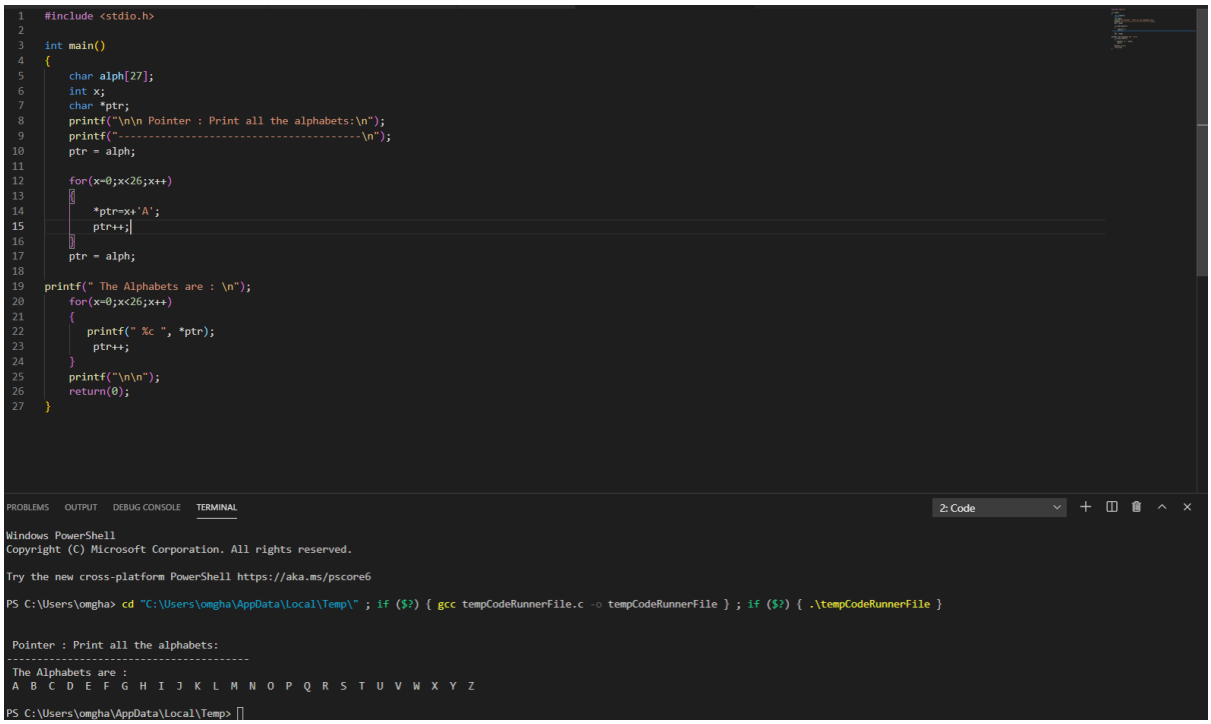
```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL                                  2: Code

Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Users\omgha> cd "C:\Users\omgha\AppData\Local\Temp\" ; if ($?) { gcc tempCodeRunnerFile.c -o tempCodeRunnerFile } ; if ($?) { .\tempCodeRunnerFile }


 Pointer : Find the factorial of a given number :
-----------------------------------------------------
 Input a number : 5
 The Factorial of 5 is : 120
```

Q.19 Write a program in C to sort an array using Pointer.

```c
#include <stdio.h>
void main()
{
   int *a,i,j,tmp,n;
    printf("\n\n Pointer : Sort an array using pointer :\n");
    printf("-------------------------------------------\n");

   printf(" Input the number of elements to store in the array : ");
   scanf("%d",&n);

   printf(" Input %d number of elements in the array : \n",n);
   for(i=0;i<n;i++)
     {
     printf(" element - %d : ",i+1);
     scanf("%d",a+i);
     }
   for(i=0;i<n;i++)
   {
    for(j=i+1;j<n;j++)
    {
       if( *(a+i) > *(a+j))
        {
       tmp = *(a+i);
       *(a+i) = *(a+j);
       *(a+j) = tmp;
        }
    }
   }
   printf("\n The elements in the array after sorting : \n");
   for(i=0;i<n;i++)
     {
     printf(" element - %d : %d \n",i+1,*(a+i));
     }
printf("\n");
}
```

```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Users\omgha> cd "C:\Users\omgha\AppData\Local\Temp\" ; if ($?) { gcc tempCodeRunnerFile.c -o tempCodeRunnerFile } ; if ($?) { .\tempCodeRunnerFile }


 Pointer : Sort an array using pointer :
 -------------------------------------------
 Input the number of elements to store in the array : 5
 Input 5 number of elements in the array :
 element - 1 : 6
 element - 2 : 5
 element - 3 : 3
 element - 4 : 8
 element - 5 : 9

 The elements in the array after sorting :
 element - 1 : 3
 element - 2 : 5
 element - 3 : 6
 element - 4 : 8
 element - 5 : 9
```

Q.20 Write a program in C to print all the alphabets using a pointer.



```c
#include <stdio.h>

int main()
{
    char alph[27];
    int x;
    char *ptr;
    printf("\n\n Pointer : Print all the alphabets:\n");
    printf("----------------------------------------\n");
    ptr = alph;

    for(x=0;x<26;x++)
    {
        *ptr=x+'A';
        ptr++;
    }
    ptr = alph;

printf(" The Alphabets are : \n");
    for(x=0;x<26;x++)
    {
       printf(" %c ", *ptr);
        ptr++;
    }
    printf("\n\n");
    return(0);
}
```

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL                                  2: Code

Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Users\omgha> cd "C:\Users\omgha\AppData\Local\Temp\" ; if ($?) { gcc tempCodeRunnerFile.c -o tempCodeRunnerFile } ; if ($?) { .\tempCodeRunnerFile }


 Pointer : Print all the alphabets:
----------------------------------------
 The Alphabets are :
 A  B  C  D  E  F  G  H  I  J  K  L  M  N  O  P  Q  R  S  T  U  V  W  X  Y  Z

PS C:\Users\omgha\AppData\Local\Temp>
```