Green University of Bangladesh

Department of Computer Science and Engineering(CSE)

Faculty of Sciences and Engineering

Semester:2nd (Summer, Year:2022), B.Sc. in CSE (Day)

Continuous Lab Performance

Course Title: Structured Programming Lab

Course Code: CSE 104 / Section: DK

Continuous Lab Performance No – 01 Solution paper

Student Details

Name	ID
Khondokar Saim	221902353

CLP Date : 23 / 08 / 2022

Submission Date : 24 / 08 / 2022

Course Teacher's Name : Monoshi Kumar Roy

[For Teachers use only: Don't Write Anything inside this box]

Marks: ……	Signature:.....
Comments:.....	Date:.....

1. Write an algorithm and implement a program in C to take an integer number from the user and print its prime digits and their sum of factorial

Solution :

Code :

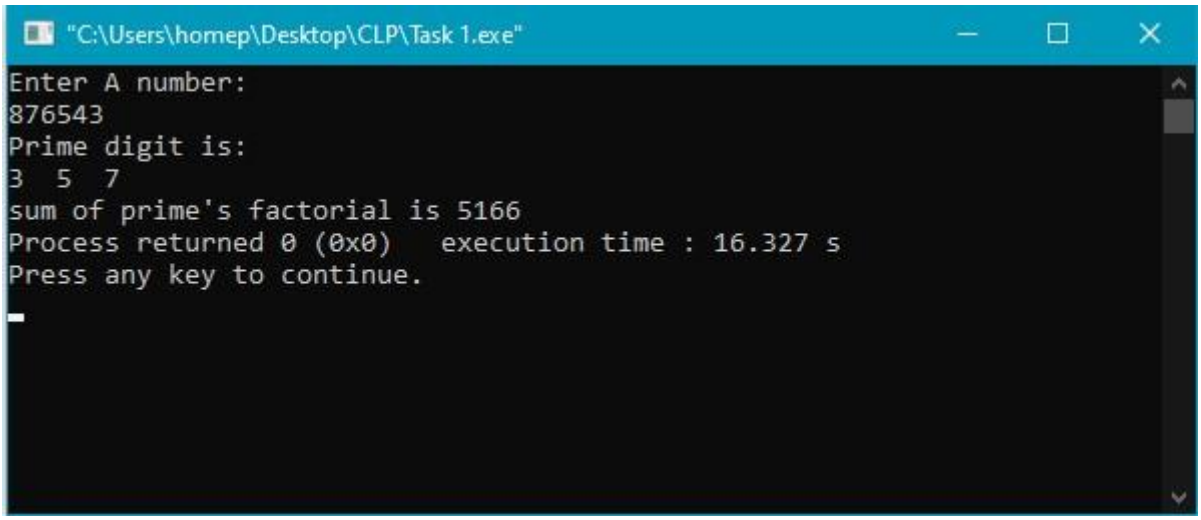
```
#include<stdio.h>
#include<math.h>
int main()
{
   int n,i,j,temp,digit,index=0,fact,sum=0,flag;
   printf("Enter A number:\n");
   scanf("%d",&n);
   digit=(int)log10(n)+1;
   int arrA[digit+1];
   while(n!=0)
      {
         temp=n%10;
         arrA[index]=temp;
         index++;
         n/=10;
      }
   printf("Prime digit is:\n");
   for(i=0;i<index;i++)
      {
         flag=0;
         if(arrA[i]==0 || arrA[i]==1)
            flag=1;
         for(j=2;j<=arrA[i]/2;j++)
```

```
{
            if(arrA[i]%j==0)
               {
                  flag=1;
                  break;
               }
           }

        if(flag==0)
           {
              printf("%d  ",arrA[i]);

              fact=1;
              for(j=1;j<=arrA[i];j++)
                 {
                    fact=fact*j;
                 }
              sum=sum+fact;
           }
      }
  printf("\nsum of prime's factorial is %d",sum);
   return 0;
}
```

Output :



2. Write an algorithm and implement a program in C to find the sum of all the digits in the odd position of a number.

Solution :

Code :

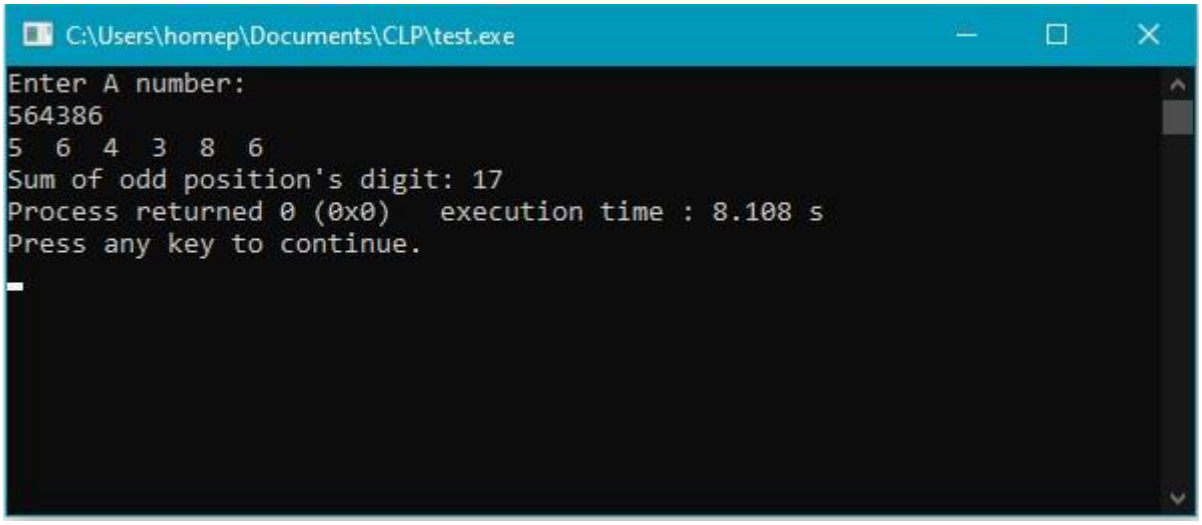
```
#include<stdio.h>
#include<math.h>
int main()
{
   int digit,n,i,temp,temp2,index=0,sum=0;
   printf("Enter A number:\n");
   scanf("%d",&n);
   digit=(int)log10(n)+1;
   int arrA[digit];

   while(n!=0)
   {
      temp=n%10;
      arrA[index]=temp;
      index++;
      n/=10;
   }

   for( i = 0; i<digit/2; i++){
      temp2 = arrA[i];
      arrA[i] = arrA[digit-i-1];
      arrA[digit-i-1] = temp2;
   }
   for(i = 0; i < digit; i++){
      printf("%d  ", arrA[i]);
   }
```

```
for(i=0;i<digit;i++)
   {
      if(i%2==0)
         {
            sum=sum+arrA[i];
         }
   }
   printf("\nSum of odd position's
digit: %d",sum);
   return 0;
}
```

Output :



3. Given an array Arr[] of integers and two numbers, n and m, perform n number of right to left and m number of left to right movements on the array.Return the updated array to be printed as a single line of space - separated integers

Solution :

Code :

```
#include<stdio.h>
int main()
{
   int i, N,temp,n,m;

   printf("Enter the size of array:\n");
   scanf("%d",&N);
   int a[N];

   printf("Enter %d integer numbers\n", N);
   for(i = 0; i < N; i++)
      scanf("%d", &a[i]);

   printf("Enter the number of positions for right to left shift\n");
   scanf("%d", &n);
    printf("Enter the number of positions for left to right shift\n");
   scanf("%d", &m);

   while(m)
   {
      if(m>0)
      {
         temp = a[0];
         for(i = 0; i < N - 1; i++)
            a[i] = a[i + 1];
```

```
a[N - 1] = temp;
      }

      m--;
   }
   while(n)
      {
         if(n>0)
      {
         temp = a[N - 1];
         for(i = N - 1; i > 0; i--)
            a[i] = a[i - 1];

         a[0] = temp;
      }
      n--;
      }
 printf("Array after shift operation ...\n");
   for(i = 0; i < N; i++)
      printf("%d  ", a[i]);

   printf("\n");
       return 0;
}
```

Output :