Q1. Write a menu driven C program to calculate area of cylinder, cube and cone.

```
#include<stdio.h>
#include<math.h>
int main() {
   int ch, yn;
   float r,h,s;
   do{
      printf("enter the shape of which u want the area to be calclulated\n");
      printf("1.Cylinder\n2.Cube\n3.Cone\n");
      printf("%d", &ch);

   switch(ch){
      case 1: printf("enter the radius and height of culinder:\n");
            scanf("%f %f", &r, &h);
            printf("The area of the cylinder is 0.2%f:\n", ((2*M_PI*r*h)+(2*M_PI*pow(r,2))));
            break;
      case 2: printf("enter the size of the cone:\n");
            scanf("%f", &s);
            printf("the area of the cone is 0.2%f:\n", (6*pow(s,2)));
            break;
      case 3: printf("enter the radius and height of the cone:\n");
            scanf("%f %f", &r, &h);
            printf("the area of the cone is 0.2%f\n", M_PI*r*(r+sqrt(pow(h,2)+pow(r,2))));
            break;
        default: printf("invalid choice");
   }
      printf("do u want to continue(1/0):\n");
      scanf("%d", &yn);
 }while(yn);
return 0;
}
```

Q2. Write a C program to check number is prime or not.

```
#include<stdio.h>
int main(){
   int i,n,flag=1;
   printf("enter the number to be checked forprime number:\n");
   scanf("%d", &n);
   for(i=2;i<n;i++){
         if (n%i==0){
            flag=0;
            break;
         }
   }
   if(flag) printf("%d is a prime number.",n);
      else printf("%d is not prime number",n);
      return 0;
}
```

Q3. Write a C program to check 3-digit number is Armstrong number or not.

```
#include<stdio.h>
#include<math.h>
int main(){
int n, temp,d=0,sum=0;
printf("Enter the number to be checked for armstrong:\n");
scanf("%d",&n);
temp=n;
while(n){
   n/=10;
   d++;
   ;
}
n=temp;
while(n){
   sum+=pow(n%10,d);
   n/=10;
}
if(sum==temp)printf("%d is an armststong number",temp);
```

```
   else printf("%d is not an armstrong number.",temp);
   return 0;
}
```

Q4. Write a program to find the list of Neon numbers

```
#include<stdio.h>
#include<math.h>
int main(){
int n, temp,d=0,sum=0;
printf("Enter the number to be checked for armstrong:\n");
scanf("%d",&n);
temp=n;
while(n){
   n/=10;
   d++;
   ;
}
n=temp;
while(n){
   sum+=pow(n%10,d);
   n/=10;
}
if(sum==temp)printf("%d is an armststong number",temp);
   else printf("%d is not an armstrong number.",temp);
   return 0;
}
```

Q5. Write a program to check whether the number is a part of Fibonacci series.

```
#include<stdio.h>
void main (){
   int num,a=0,b=1,c=1, ispart=0;
   printf("Enter a number:\n");
   scanf("%d",&num);
   while (c<=num){
      c=a+b;
      a=b;
      b=c;
      if(c==num){
         ispart=1;
          break;


      }
   }
   if(ispart)
      printf("%d is a part of a fibocnacci series", num);
      else
      printf("%d is nota part of fibonacci seriess",num);


}
```

Q6. Write a C program to find factorial of number using function.

```
#include<stdio.h>
 unsigned long int fact(int n);
 int main(){
   int n;
   printf("enter the number:\n");
   scanf("%d", &n);
   printf("the fact of the number of %d is %lu.",n,fact(n));
   return 0;
}
unsigned long int fact(int n){
   int i,fact=1;
   for(i=1;i<=n;i++){
      fact *= i;
   }
   return fact;
   }
```

Q7. Write a C program to display pattern (any pattern)

```
#include <stdio.h>

int main() {
   int rows, i, j;

   printf("Enter the number of rows: ");
   scanf("%d", &rows);

   for (i = 1; i <= rows; i++) {
      // Print spaces for right-angled triangle (optional)
      for (int space = 1; space <= rows - i; space++) {
         printf("*");
      }

      // Print asterisks
      for (j = 1; j <= i; j++) {
         printf(" ");
      }
      printf("\n");
   }

   return 0;
}
```

Q8. Write a program to accept ‘n’ integers from the user into an array and search an element using linear search method.

```
#include<stdio.h>
int main() {
int n, i, x, xInd=-1;
printf("Enter number of array elements:\n"); scanf("%d", &n);
printf("Enter Array Elements: \n");
int arr[n];
for(i=0;i<n;i++) scanf("%d", &arr[i]);
printf("Enter Element to be Searched:\n");
scanf("%d", &x);
for(i=0;i<n;i++) {
if (arr[i] == x ) {
xIndi+1;
break;
宮
}
}
if (xInd == -1) printf("\nElement not found.");
else printf("Element is found at %d.", xInd); return 0;
}
```

Q9. Write a C program to convert decimal number into binary number using array.

```
#include <stdio.h>

int main() {
   int decimal, i, remainder;
   int binary[32];  // Array to store binary digits (maximum 32 bits)

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

   // Initialize array elements to 0
   for (i = 0; i < 32; i++) {
      binary[i] = 0;
   }

   i = 0;
   while (decimal > 0) {
      remainder = decimal % 2;
      binary[i] = remainder;
      decimal /= 2;
      i++;
   }

   printf("Binary equivalent: ");
```

```
    // Print binary representation in reverse order (most significant bit last)
    for (i = 31; i >= 0; i--) {
        printf("%d", binary[i]);
    }
    printf("\n");

    return 0;
}
```

Q10. Write a c program to find nCr and nPr using function.

```
#include<stdio.h>

long long int factorial(int n){
    if(n==0){
        return 1;
    }
    else{
        return n* factorial(n-1);
    }
}

long long int nCr(int n, int r){
if(r>n){
    return 0;
} else{
    return factorial(n) / (factorial(r) * factorial(n-r));
    }
}

long long int nPr(int n, int r){
if(r>n){
    return 0;

}else {
return factorial(n)/(factorial(n-r));
}
}

int main(){
int n,r;
printf("enter the value of n and r:\n");
scanf("%d %d", &n, &r);
printf("nCr= %lld\n", nCr(n,r));
printf("nPr=%lld\n",nPr(n,r));
return 0;

}
```

Q11. Write a c program to find factorial using recursion

```
#include<stdio.h>
long long int factorial(int n){
if (n==0){
    return 1;
}
else{
    return n*factorial(n-1);
}
}

int main(){

int num;
printf("enter a non negative number:");
scanf("%d",&num);
if (num<0){
    printf("error");
    return 1;
}

long long int result =factorial(num);
printf("factorial of %d is %lld\n", num , result);
return 0;
```

```
}
```

Q12. Write a c program to find GCD using recursion.

```
#include<stdio.h>
int gcd(int a,int b);
int main(){
   int a,b;
   printf("Enter 2 numbers:\n");
   scanf("%d %d", &a,&b);
   printf("the gcd of the 2 number is %d",gcd(a,b));
   return 0;
}
int gcd(int a,int b){
if (b==0) return a;
return gcd(b,a%b);
}
```

Q13. Write a c program to find power of number using recursion.

```
#include<stdio.h>
long int power(int x,int y);
int main(){
int x,y;
printf("eter the base and power:\n");
scanf("%d %d", &x, &y);
printf("%d raise to the power %d is %ld",x,y,power(x,y));
return 0;

}
long int power(int x,int y){
if(y==0) return 1;
return x*power(x,y-1);}
```

Q14. Write a c program to add two matrices.

```
#include<stdio.h>
#define MAX_rows 100
#define Max_cols 100

int main(){
    int rows,cols;
    printf("enter the no. of rows and cols of the matrice (max %d x %d):",MAX_rows,Max_cols);
    scanf("%d %d", &rows, &cols);
    if (rows<=0||cols<0||rows>MAX_rows||cols>Max_cols){
      printf("invalid");
      return 1;
    }

    int matrix1[MAX_rows][Max_cols], matrix2 [MAX_rows][Max_cols], sum[MAX_rows][Max_cols];

      printf("enter the ele of matrix1:\n");
      for(int i=0;i<rows;i++){
         for(int j=0;j<cols;j++){
            scanf("%d", &matrix1[i][j]);
         }
      }
       printf("enter the ele of matrix2:\n");
      for(int i=0;i<rows;i++){
         for(int j=0;j<cols;j++){
            scanf("%d", &matrix2[i][j]);
         }
      }
       for(int i=0;i<rows;i++){
         for(int j=0;j<cols;j++){
               sum[i][j]= matrix1[i][j] + matrix2[i][j];
         }}

         printf("sum of mat:\n");
         for(int i=0;i<rows;i++){
         for(int j=0;j<cols;j++){
               printf("%d\t", sum[i][j]);


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

```
        }
        return 0;
}
```

Q15. Write a c program to subtract two matrices

```
#include<stdio.h>
#define MAX_rows 100
#define Max_cols 100

int main(){
    int rows,cols;
    printf("enter the no. of rows and cols of the matrice (max %d x %d):",MAX_rows,Max_cols);
    scanf("%d %d", &rows, &cols);
    if (rows<=0||cols<0||rows>MAX_rows||cols>Max_cols){
      printf("invalid");
      return 1;
    }

    int matrix1[MAX_rows][Max_cols], matrix2 [MAX_rows][Max_cols], sub[MAX_rows][Max_cols];

      printf("enter the ele of matrix1:\n");
      for(int i=0;i<rows;i++){
        for(int j=0;j<cols;j++){
          scanf("%d", &matrix1[i][j]);
        }
      }
       printf("enter the ele of matrix2:\n");
      for(int i=0;i<rows;i++){
        for(int j=0;j<cols;j++){
          scanf("%d", &matrix2[i][j]);
        }
      }
      for(int i=0;i<rows;i++){
        for(int j=0;j<cols;j++){
            sub[i][j]= matrix1[i][j]-matrix2[i][j];
        }}

       printf("sub of mat:\n");
        for(int i=0;i<rows;i++){
        for(int j=0;j<cols;j++){
            printf("%d\t", sub[i][j]);


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

Q16. . Write a c program to multiply two matrices

```
#include<stdio.h>
#define MAX_rows 100
#define Max_cols 100

int main(){
    int rows,cols;
    printf("enter the no. of rows and cols of the matrice (max %d x %d):",MAX_rows,Max_cols);
    scanf("%d %d", &rows, &cols);
    if (rows<=0||cols<0||rows>MAX_rows||cols>Max_cols){
      printf("invalid");
      return 1;
    }

    int matrix1[MAX_rows][Max_cols], matrix2 [MAX_rows][Max_cols], mul[MAX_rows][Max_cols],k;

      printf("enter the ele of matrix1:\n");
      for(int i=0;i<rows;i++){
        for(int j=0;j<cols;j++){
          scanf("%d", &matrix1[i][j]);
        }
```

```
    }
     printf("enter the ele of matrix2:\n");
    for(int i=0;i<rows;i++){
       for(int j=0;j<cols;j++){
          scanf("%d", &matrix2[i][j]);
       }
    }
    for(int i=0;i<rows;i++){
       for(int j=0;j<cols;j++){
             mul[i][j]=0;
             for(k=0;k<cols;k++){
             mul[i][j]+= matrix1[i][k]*matrix2[k][j];
       }}}

     printf("mul of mat:\n");
       for(int i=0;i<rows;i++){
       for(int j=0;j<cols;j++){
             printf("%d\t", mul[i][j]);


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

Q17. Write a c program to find transpose of a matrix with and without new array

a. With New Array

```
#include<stdio.h>
#define MAX_rows 100
#define Max_cols 100

int main(){
    int rows,cols;
    printf("enter the no. of rows and cols of the matrice (max %d x %d):",MAX_rows,Max_cols);
    scanf("%d %d", &rows, &cols);
    if (rows<=0||cols<0||rows>MAX_rows||cols>Max_cols){
      printf("invalid");
      return 1;
    }

    int matrix1[MAX_rows][Max_cols], matrix2 [MAX_rows][Max_cols], mul[MAX_rows][Max_cols],k;

      printf("enter the ele of matrix1:\n");
      for(int i=0;i<rows;i++){
         for(int j=0;j<cols;j++){
            scanf("%d", &matrix1[i][j]);
         }
      }
       printf("enter the ele of matrix2:\n");
      for(int i=0;i<rows;i++){
         for(int j=0;j<cols;j++){
            scanf("%d", &matrix2[i][j]);
   }

      }
      for(int i=0;i<rows;i++){
         for(int j=0;j<cols;j++){
               mul[i][j]=0;
               for(k=0;k<cols;k++){
               mul[i][j]+= matrix1[i][k]*matrix2[k][j];
         }}}

       printf("mul of mat:\n");
         for(int i=0;i<rows;i++){
         for(int j=0;j<cols;j++){
               printf("%d\t", mul[i][j]);


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

```
}
```

b. Without new Array.

```
#include<stdio.h>
int main() {
int i,j,n,m, temp;
printf("Enter the order of first matrix (n,m):\n"); scanf("%d %d", &n, &m);
printf("Enter first array elements:\n");
int a[n] [m];
for(i=0;i<n;i++) for(j=0;j<m;j++) scanf("%d" ,&a[i][j]);
if (n!=m) {
printf("Invalid Matrix.");
return 0;
} else {
for(i=0;i<n;i++) {
for(j=0;j<m;j++) {
if (j>i) { temp = a[i][j]; a[j] [i] = temp; }
a[i][j] = a[j] [i];
}
}
}
for(i=0;i<n;i++) {
for(j=0;j<m;j++) {
printf("%d\t", a[i][j]);
}
printf("\n");
}
return 0;
}
```

Q18. Write a c program to check matrix is symmetrical or not.

```
#include<stdio.h>

int main(){
int rows,cols,issym=1,i,j;
printf("enter the no. of rows and cols:\n");
scanf("%d %d", &rows, &cols);
if(rows!=cols){
   printf("not square matrix, symmetry defined only for square matrix.");
   return 1;
}
int matrix[rows][cols];
printf("enter elements of matrix:\n");
for(int i=0;i<rows;i++){
   for(int j=0;j<cols;j++){
      scanf("%d", &matrix[i][j]);
   }
}
for (i=0;i<rows && issym;i++){
   for (j=0;j<cols && issym;j++){
      if(matrix[i][j] != matrix[i][j]){
         issym=0;
         break;
      }
   }
}

if(issym){
   printf("yes syym.");
}else{
printf("no not syym.");}
return 0;
}
```

Q19. Write a c program to sort an array list in ascending order using bubble sort method.

```
#include<stdio.h>
int main(){
   int n,temp,i,j;
   printf("Enter the size of array:\n");
   scanf("%d", &n);
   int arr[n];
   printf("Enter the array elements:\n");
   for(i=0;i<n;i++){scanf("%d", &arr[i]);}

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

Q20. Write a program that checks whether the entered string is a palindrome or not, without using string header file

```
#include<stdio.h>
int main(){
   int n,temp,i,j;
   printf("Enter the size of array:\n");
   scanf("%d", &n);
   int arr[n];
   printf("Enter the array elements:\n");
   for(i=0;i<n;i++){scanf("%d", &arr[i]);}

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

```
}
```

Q21. Write a program to insert an element into array using function.

```
#include<stdio.h>
void insertelement(int arr[],int *n, int posi, int x);
int main(){
   int n,i,posi,x;
   printf("Enter the size of array:\n");
   scanf("%d", &n);
   int arr[n];
   printf("Enter array elements:\n");
   for (i=0;i<n;i++) scanf("%d", &arr[i]);
   printf("enter the elements to be inserted");
   scanf("%d",&x);
   printf("enter the position to be inserted");
   scanf("%d", &posi);
   insertelement(arr,&n,posi,x);
   printf("Array after insertion :\n");
   for(i=0;i<n;i++)printf("%d", arr[i]);
   return 0;
}
void insertelement(int arr[],int *n, int posi, int x){
   int i;
   for(i=*n;i>posi;i--){
      arr[i]=arr[i-1];
   }
   arr[i]=x;
   (*n)++;
}
```

Q22. Write a program to store and display the name, roll number, and total PCM marks for one Student using Structure

```
#include<stdio.h>
struct student{
   char name[50];
   int rno;
   float phy,chem,math,total;
};

int main(){
   struct student s;
   printf("Enter student name:");
   scanf("%s", &s.name);

   printf("enter roll.no");
   scanf("%d",&s.rno);
   printf("entr marks phy");
   scanf("%f",&s.phy);
   printf("entr chem marks");
   scanf("%f",&s.chem);
   printf("enter math marks");
   scanf("%f",&s.math);
   s.total=s.phy+s.math+s.chem;
   printf("name:%s\n", s.name);
   printf("rollno:%d\n", s.rno);
   printf("phy marks:%0.f\n", s.phy);
   printf("chem marks:%0.f\n", s.chem);
   printf("math marks:%0.f\n", s.math);
   printf("total:%f\n",s.total);
   return 0;
}
```

Q23. Write a c program to store and display the employee code, name, date of joining, salary and department for one employee using structure. Use Nested structure for date of Joining.

```
#include<stdio.h>
struct employee{
   char name[100],dept[100];
```

```
   int code;
   float salary;
   struct doj{
      int d,m,y;
      }date;
};

int main(){
   struct employee e;
   printf("Enter employe code.:");
   scanf("%d", &e.code);
   printf("enter employee name.:");
   scanf("%s", &e.name);
   printf("enter employee dept");
   scanf("%s", &e.dept);
   printf("enter date of joining");
   scanf(" %d %d %d",&e.date.d,&e.date.m,&e.date.y);
   printf("Enter employee salary");
   scanf("%f", &e.salary);
   printf("\n");
   printf("Employee code.:%d\n", e.code);
   printf("Employee name.:%s\n",e.name);
   printf("Employee department.:%s\n",e.dept);
   printf("Date of joining.: %d %d %d\n",e.date.d , e.date.m , e.date.y );
   printf("Employee salary.:%f", e.salary);
   return 0;
}
```

Q24. Write a c program to display month name in word when enter a number using switch case.

```
#include<stdio.h>
int main(){
   int n;
   printf("Enter a number to display its corresponding month:");
   scanf("%d", &n);
   switch(n){
   case(1): printf("january");
         break;
   case(2): printf("feb");
         break;
   case(3): printf("march");
         break;
   case(4): printf("april");
         break;
    case(5): printf("may");
         break;
   case(6): printf("june");
         break;
   case(7): printf("july");
         break;
   case(8): printf("august");
         break;

   case(9): printf("sept");
         break;
    case(10): printf("oct");
         break;
    case(11): printf("NOv");
         break;
    case(12): printf("dec");
         break;
     default: printf("Invalid Input.");
   }
   return 0;
}
```

Q25. Write a c program to find sum of diagonal elements of a matrix

```
#include<stdio.h>
int main(){
   int n,m,i,j,mainsum=0,offsum=0;
   printf("enter the order of matrix(n,m):");
   scanf("%d %d", &n,&m);
```

```
    int arr[n][m];
    if(n!=m){
        printf("not a square matrix.");
        return 0;
    }
    printf("enter matrix elements:\n");
    for(i=0;i<n;i++) for (j=0;j<m;j++) scanf("%d",&arr[i][j]);
    for(i=0;i<n;i++){
        mainsum+=arr[i][i];
    }
    i=0;j=m-1;
    while(i<n){
        offsum += arr[i][j];
        i++,j--;

    }
    printf("the sum of main diagonal elements is %d\n", mainsum);
    printf("the sum of off diagonal element is %d\n",offsum);
return 0;
}
```

Q26. Write a program to accept ‘n’ integers from the user into an array and find display even number and prime number.

```
#include <stdio.h>

int is_prime(int num) {
  if (num <= 1) {
    return 0; // 1 or less are not prime
  }
  for (int i = 2; i * i <= num; i++) {
    if (num % i == 0) {
      return 0; // Divisible by other numbers, not prime
    }
  }
  return 1; // Prime number
}

int main() {
  int n, i;

  printf("Enter the number of elements (n): ");
  scanf("%d", &n);

  int arr[n];

  printf("Enter %d integers:\n", n);
  for (i = 0; i < n; i++) {
    scanf("%d", &arr[i]);
  }

  printf("Even numbers:\n");
  for (i = 0; i < n; i++) {
    if (arr[i] % 2 == 0) {
      printf("%d ", arr[i]);
    }
  }
  printf("\n");

  printf("Prime numbers:\n");
  for (i = 0; i < n; i++) {
    if (is_prime(arr[i])) {
      printf("%d ", arr[i]);
    }
  }
  printf("\n");

  return 0;
}
```

Q27. Write a program to accept ‘n’ integers from the user into an array and find the smallest and largest integer.

```
#include<stdio.h>
int main(){
   int n,i,maxi,mini;
   printf("enter array size:");
   scanf("%d", &n);
   int arr[n];
   printf("enter array elements");
   for(i=0;i<n;i++) scanf("%d",&arr[i]);
   maxi=mini=arr[0];
   for(i=0;i<n;i++){
      if(arr[i]>maxi) maxi =arr[i];
      else if (arr[i]<mini) mini =arr[i];
   }
   printf("smallest %d biggest %d", mini,maxi);
   return 0;
}
```

Q28. Write a C program to compare two string using strcmp() function.

```
#include<stdio.h>
#include<string.h>
int main(){
   char s1[100],s2[100];
   int r;
   printf("enter string 1");
   gets(s1);
   printf("Enter string 2");
   gets(s2);
   r=strcmp(s1,s2);
   if(r==0)printf("both are same");
   else if(r>0)printf("Str 1 is big");
   else printf("Str 2 is big");
   return 0;

}
```

Q29. Write a C program to concatenate two strings without using string.h.

```
#include<stdio.h>
void ConcatenateString(char str1[100],char str2 [100],char res [200]){
   int i=0,j=0;
   while(str1[i]!= '\0'){
      res[i]=str1[i];
      i++;
   }
   while(str2[j]!='\0'){
      res[i+j]=str2[j];
      j++;
   }
   res[i+j]='\0';
}
int main(){
      char str1[100],str2 [100],res [200];
      printf("Enter str1");
      gets(str1);
      printf("Enter str2");
      gets(str2);
      ConcatenateString(str1,str2,res);
      printf("Concated string: %s\n", res);
      return 0;
}
```

Q30. Write a c program to find sum of each row elements of a matrix.

```
#include<stdio.h>
int main(){
   int n,m,i,j,rsum=0;
   printf("enter row[n] and column[m] of matrix");
   scanf("%d %d ", &n,&m);
   int arr[n][m];
```

```
    printf("Enter the elements of matirx");
    for(i=0;i<n;i++){
        for(j=0;j<m;j++){
            scanf("%d", &arr[i][j]);
        }
    }
    printf("Summation of rows:\n");
    for(i=0;i<n;i++){
            rsum=0;
        for(int j=0;j<m;j++){
            rsum += arr[i][j];
        }
        printf("%d \t", rsum);
        rsum = 0;
    }
    return 0;
}
```