Name: Sajib Ahmed

W3 resources

C Basic Declarations and

ExpressionsExercise (1-50)

Problem 01 :

Write a C program to print your name, date of birth, and mobile number.

Solution :

```
#include <stdio.h>
int main() {

    printf("Name   : Alexandra Abramov\n");
printf("DOB    : July 14, 1975\n");
printf("Mobile : 99-9999999999\n");


    return 0;
}
```

Expected Output :

```
Name   : Alexandra Abramov
DOB    : July 14, 1975
Mobile : 99-9999999999

Process returned 0 (0x0)   execution time : 0.016 s
Press any key to continue.
```

Problem 02 :

Write a C program to get the C version you are using.

Solution :

```
#include <stdio.h> int main(int

argc, char** argv) {

   #if __STDC_VERSION__ >=  201710L
      printf("We are using C18!\n");
   #elif __STDC_VERSION__ >= 201112L
      printf("We are using C11!\n");
   #elif __STDC_VERSION__ >= 199901L
      printf("We are using C99!\n");
   #else
      printf("We are using C89/C90!\n");
   #endif
   return 0;
}
```

Expected Output :

```
PS E:\Code for VS> cd "e:\Code for VS\" ; if ($?) { gcc ex.c -o ex } ; if ($?) { .\ex }
We are using C18!
PS E:\Code for VS>
```

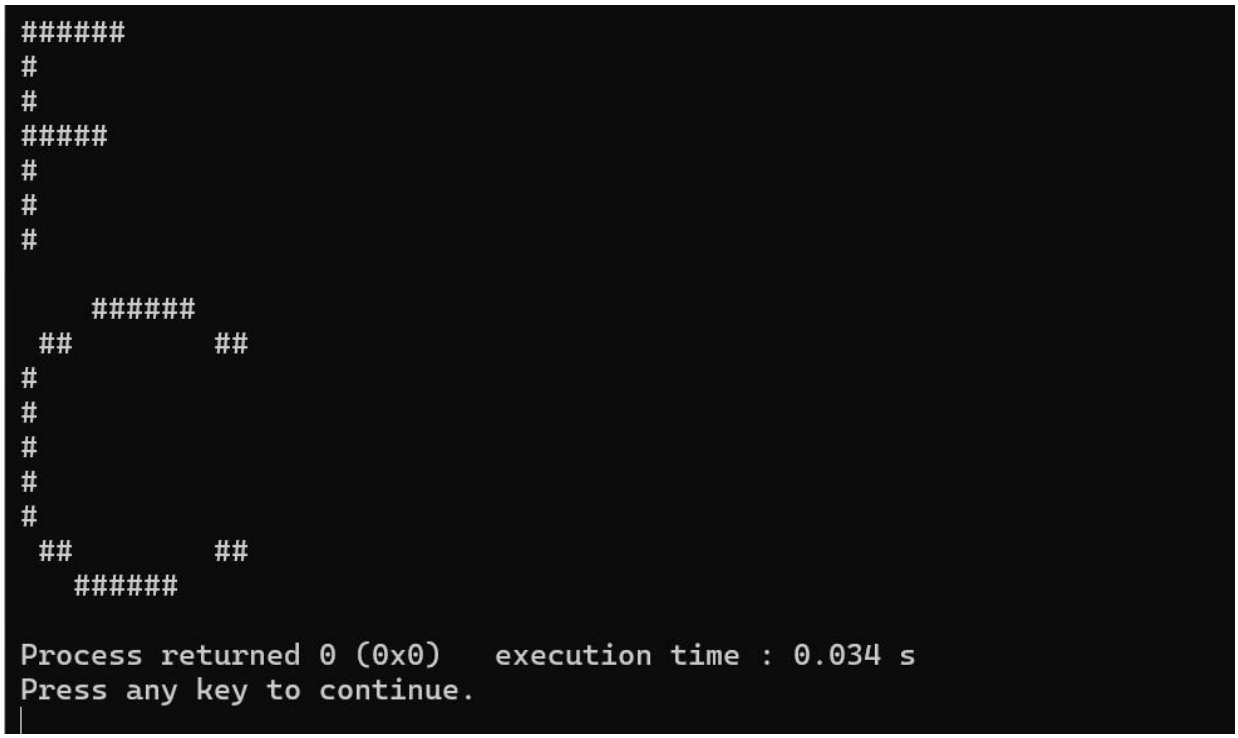
Problem 03 :

Write a C program to print a block F using the hash (#), where the F has a height of seven characters and width of five and four characters. And also print a very large 'C'.

Solution :

```
#include <stdio.h>
int main() {

   printf("######\n");
printf("#\n");
printf("#\n");
printf("#####\n");
printf("#\n");
printf("#\n");
printf("#\n");
printf("\n");

   printf("    ######\n");
printf(" ##        ##\n");
printf("#\n");
printf("#\n");
printf("#\n");
printf("#\n");
printf("#\n");     printf("
##        ##\n");     printf("
######\n");

  return 0;
}
```

Expected Output :



Problem 04 :

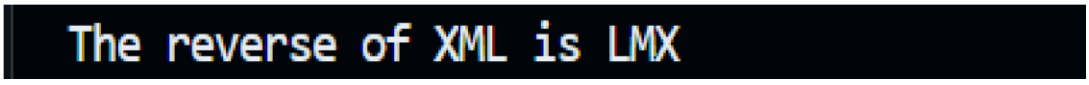
Write a C program to print the following characters in reverse.

Test Characters: 'X', 'M', 'L'

Solution :

```
#include <stdio.h>

int main(){

  char c1='X';   char c2='M';   char c3='L';   printf("The
reverse of XML is %c%c%c\n",c3,c2,c1);

    return 0;
}
```

Expected Output :



Problem 05:

Write a C program to compute the perimeter and area of a rectangle with a height of 7 inches and width of 5 inches.

Solution:

```
#include <stdio.h>

int main(){

 int height = 7;
int width = 5;

 int perimeter = 2*(height+width);
int area = height * width;


  printf("Perimeter of the rectangle = %d inches\n",perimeter);
printf("Area of the rectangle = %d square inches\n",area);

   return 0;
}
```

Expected Output :

```
Perimeter of the rectangle = 24 inches
Area of the rectangle = 35 square inches

Process returned 0 (0x0)   execution time : 0.021 s
Press any key to continue.
```

Problem 06:

Write a C program to compute the perimeter and area of a circle with a given radius of 6 inches ***Solution :***

```
#include <stdio.h>
#define pi 3.1416
int main(){

  float r = 6;   float
perimeter, area;
perimeter = 2 * pi * r;
area = pi * r * r;
  printf("Perimeter of the Circle = %10f inches\n",perimeter);
printf("Area of the Circle = %10f square inches\n",area);

    return 0;
}
```

Expected Output :

```
Perimeter of the Circle =  37.699200 inches
Area of the Circle = 113.097603 square inches

Process returned 0 (0x0)   execution time : 0.032 s
Press any key to continue.
```

Problem 07 :

Write a C program to display multiple variables.

Sample *Variable*s : a+ c, x + c,dx + x, ((int) dx) + ax, a + x, s + b, ax + b, s + c, ax + c, ax + ux

Solution :

```
#include <stdio.h>

int main()
{
   int a = 125, b = 12345;
long ax = 1234567890;
short s = 4043;
float x = 2.13459;
double dx = 1.1415927;
char c = 'W';
   unsigned long ux = 2541567890;

   printf("a + c =  %d\n", a + c);
printf("x + c = %f\n", x + c);
printf("dx + x = %f\n", dx + x);
   printf("((int) dx) + ax =  %ld\n", ((int)
dx) + ax);     printf("a + x = %f\n", a +
x);     printf("s + b =  %d\n", s + b);
printf("ax + b = %ld\n", ax + b);
printf("s + c =  %hd\n", s + c);
printf("ax + c = %ld\n", ax + c);
printf("ax + ux = %lu\n", ax + ux);

   return 0;
}
```

Expected Output :

```
a + c =  212
x + c = 89.134590
dx + x = 3.276183
((int) dx) + ax =  1234567891
a + x = 127.134590
s + b =  16388
ax + b = 1234580235
s + c =  4130
ax + c = 1234567977
ax + ux = 3776135780
```

Problem 08 :

Write a C program to convert specified days into years, weeks and days.

Note: Ignore leap year. ***Solution :***

```
#include

<stdio.h> int

main() {

   int years, weeks, days, day;
days = 1329;     years =
days/365;     weeks =
(days%365)/7;     day = days -
((years*365)+(weeks*7));

   printf ("Years :
%d\n",years);     printf
("Weeks : %d\n",weeks);
printf ("Days : %d\n",day);



   return 0;
}
```

Expected Output :

```
Years : 3
Weeks : 33
Days : 3
```

Problem 09 :

Write a C program that accepts two integers from the user and calculates the sum of the two integers.

Solution:

```
#include <stdio.h>
int main()
{
   int x, y, sum;

   printf("Input the first integer: ");
scanf("%d", &x);

   printf("Input the second integer: ");    scanf("%d",
&y);    sum = x + y;    printf("Sum of the above two
integers = %d\n", sum);

   return 0;
}
```

Expected Output :

```
Input the first integer: 5
Input the second integer: 7
Sum of the above two integers = 12
```

Problem 10 :

Write a C program that accepts two integers from the user and calculates the product of the two integers. ***Solution :***

```
#include <stdio.h>

int main()
{
   int x, y, product;

   printf("Input the first integer: ");
scanf("%d", &x);


   printf("Input the second integer: ");
scanf("%d", &y);     product = x * y;

   printf("Product of the above two integers = %d\n", product);
}
```

Expected Output :

```
Input the first integer: 5
Input the second integer: 7
Product of the above two integers = 35
```

Problem 11:

Write a C program that accepts two item's weight and number of purchases (floating point values) and calculates their average value.

Solution:

```
#include <stdio.h>

int main()
{
   double wi1, ci1, wi2, ci2, result;
   printf("Weight - Item1: ");
   scanf("%lf", &wi1);

   printf("No. of item1: ");
   scanf("%lf", &ci1);

   printf("Weight - Item2: ");
   scanf("%lf", &wi2);

   printf("No. of item2: ");
   scanf("%lf", &ci2);

   result = ((wi1 * ci1) + (wi2 * ci2)) / (ci1 + ci2);

   printf("Average Value = %f\n", result);
   return 0;
}
```

Expected Output :

```
PS E:\Code for VS> cd "e:\Code for VS\" ; if ($?) { gcc ex.c -o ex } ; if ($?) { .\ex }
Weight - Item1: 15
No. of item1: 5
Weight - Item2: 25
No. of item2: 4
Average Value = 19.444444
PS E:\Code for VS>
```

Problem 12:

Write a C program that accepts an employee's ID, total worked hours in a month and the amount he received per hour. Print the ID and salary (with two decimal places) of the employee for a particular month.

Solution :

```
#include <stdio.h>
int main() {
    char id[10];
    int hour;
    double value, salary;
    printf("Input the Employees ID(Max. 10 chars): ");
    scanf("%s", &id);

    printf("\nInput the working hrs: ");
    scanf("%d", &hour);

    printf("\nSalary amount/hr: ");
    scanf("%lf", &value);

    salary = value * hour;

    printf("\nEmployees ID = %s\nSalary = U$ %.2lf\n", id, salary);

    return 0;
}
```

Expected Output :

```
PS E:\Code for VS> cd "e:\Code for VS\" ; if ($?) { gcc ex.c -o ex } ; if ($?) { .\ex }
Input the Employees ID(Max. 10 chars): 0342

Salary amount/hr: 15000

Employees ID = 0342
Salary = U$ 120000.00
```

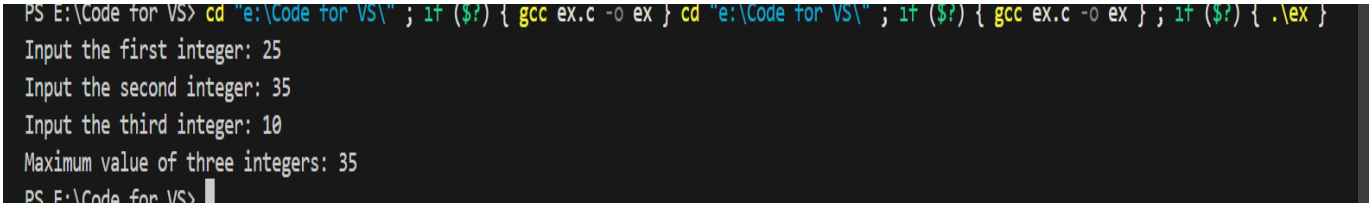
Problem 13:

Write a C program that accepts three integers and finds the maximum of three.

Solution :

```
#include <stdio.h>
int main() {

    int n, m, o;
    int product;
    printf("Input the first integer: ");
     scanf("%d",&n);

    printf("Input the second integer: ");
     scanf("%d",&m);

     printf("Input the third integer: ");
     scanf("%d",&o);

     if(m>n && m>o){
        printf("Maximum value of three integers: %d\n",m);
     }
     else if(o>n && o>m){
        printf("Maximum value of three integers: %d\n",o);
     }
    else if(n>m && n>o) {
        printf("Maximum value of three integers: %d\n",n);
     }
     else{
       printf("No valid input");
     }
      return 0;}
```

Expected Output :



Problem 14:

Write a C program to calculate a bike's average consumption from the given total distance (integer value) travelled (in km) and spent fuel (in litters, float number – 2 decimal points).

Solution :

```
#include <stdio.h>

int main()
{
  int n;
  printf("Input total distance in km: ");
   scanf("%d",&n);

  float f;
  printf("Input total fuel spent in liters: ");
   scanf("%f",&f);

   printf("Average consumption (km/It) %.3f",n/f);

}
```

Expected Output :

```
PS E:\Code for VS> cd "e:\Code for VS\" ; if ($?) { gcc ex.c -o ex } ; if ($?) { .\ex }
Input total distance in km: 350
Input total fuel spent in liters: 5
Average consumption (km/It) 70.000
```

Problem 15:

Write a C program to calculate the distance between two points.

Solution :

```
#include <stdio.h>
#include <math.h>
int main()
{
   float d;
  float x1;
  printf("Input x1: ");
   scanf("%f",&x1);
    float y1;
      printf("Input y1: ");
       scanf("%f",&y1);
    float x2;
      printf("Input x2: ");
       scanf("%f",&x2);
     float y2;
      printf("Input y2: ");
       scanf("%f",&y2);
    d = sqrt((x2-x1)*(x2-x1)+(y2-y1)*(y2-y1));

   printf("Distance between the said points  %.4f\n",d);

}
```

Expected Output :

```
Input x1: 25
Input y1: 10
Input x2: 35
Input y2: 16
Distance between the said points  11.6619
```

Problem 16:

Write a C program to read an amount (integer value) and break the amount into the smallest possible number of bank notes.

Solution :

```
#include <stdio.h>
int main() {
    int amt, total;
    printf("Input the amount: ");
    scanf("%d", &amt);
    total = amt / 100;
    printf("There are:\n");
    printf("%d Note(s) of 100.00\n", total);
    amt = amt - (total * 100);
    total = amt / 50;
    printf("%d Note(s) of 50.00\n", total);
    amt = amt - (total * 50);
    total = amt / 20;
    printf("%d Note(s) of 20.00\n", total);
    amt = amt - (total * 20);
    total = amt / 10;
    printf("%d Note(s) of 10.00\n", total);
    amt = amt - (total * 10);
    total = amt / 5;
    printf("%d Note(s) of 5.00\n", total);
    amt = amt - (total * 5);
    total = amt / 2;
    printf("%d Note(s) of 2.00\n", total);
    amt = amt - (total * 2);
    total = amt / 1;
    printf("%d Note(s) of 1.00\n", total);
    return 0;}
```

Expected Output :

```
Input the amount: 375
There are:
3 Note(s) of 100.00
1 Note(s) of 50.00
1 Note(s) of 20.00
0 Note(s) of 10.00
1 Note(s) of 5.00
0 Note(s) of 2.00
0 Note(s) of 1.00
```

Problem 17:

Write a C program to convert a given integer (in seconds) to hours, minutes and seconds.

Solution :

```
#include <stdio.h>

int main() {
   int s;
   printf("Input seconds: ");
   scanf("%d",&s);

    printf("H:M:S - %d:%d:%d ",s/3600,((s%3600)/60),((s%
3600)%60));



   return 0;
}
```

Expected Output :

```
PS E:\Code for VS> cd "e:\Code for VS\" ; if ($?) { gcc ex.c -o ex } ; if ($?) { .\ex }
Input seconds: 25300
H:M:S - 7:1:40
```

Problem 18:

Write a C program to convert a given integer (in days) to years, months and days, assuming that all months have 30 days and all years have 365 days.

Solution :

```
#include <stdio.h>

int main() {
   int d;
   printf("Input no. of days: ");
   scanf("%d",&d);

     printf("%d Year(s)\n",d/365);
     printf("%d Month(s)\n",(d%365)/30);
     printf("%d Day(s)\n",(d%365)%30);


    return 0;
}
```

Expected Output :

```
Input no. of days: 2535
6 Year(s)
11 Month(s)
15 Day(s)
```

Problem 19:

Write a C program that accepts 4 integers p, q, r, s from the user where q, r and s are positive and p is even. If q is greater than r and s is greater than p and if the sum of r and s is greater than the sum of p and q print "Correct values", otherwise print "Wrong values".

Solution :

```
#include <stdio.h>
int main() {
   int p, q, r, s;
   printf("\nInput the first integer: ");
   scanf("%d", &p);
   printf("\nInput the second integer: ");
   scanf("%d", &q);
   printf("\nInput the third integer: ");
   scanf("%d", &r);
   printf("\nInput the fourth integer: ");
   scanf("%d", &s);
   if((q > r) && (s > p) && ((r+s) > (p+q)) && (r > 0) && (s > 0)
&& (p%2 == 0))
   {
      printf("\nCorrect values\n");
   }
   else {
      printf("\nWrong values\n");
   }
   return 0;
}
```

Expected Output :

```
Input the third integer: 15

Input the fourth integer: 46

Wrong values
PS E:\Code for VS> cd "e:\Code for VS\" ; if ($?) { gcc ex.c -o ex } ; if ($?) {
```

Problem 20:

Write a C program to print the roots of Bhaskara's formula from the given three floating numbers. Display a message if it is not possible to find the roots.

Solution :

```
#include <stdio.h>
#include <math.h>

int main() {
   double a, b, c, pr1;
   printf("\nInput the first number(a): ");
   scanf("%lf", &a);
   printf("\nInput the second number(b): ");
   scanf("%lf", &b);
   printf("\nInput the third number(c): ");
   scanf("%lf", &c);
   pr1 = (b*b) - (4*(a)*(c));
   if(pr1 > 0 && a != 0) {
      double x, y;
      pr1 = sqrt(pr1);
      x = (-b + pr1)/(2*a);
      y = (-b - pr1)/(2*a);
      printf("Root1 = %.5lf\n", x);
      printf("Root2 = %.5lf\n", y);
   }
   else {
      printf("\nImpossible to find the roots.\n");
   }
   return 0;}
```

Expected Output :

```
Input the first number(a): 25

Input the second number(b): 35

Input the third number(c): 12
Root1 = -0.60000
Root2 = -0.80000
```

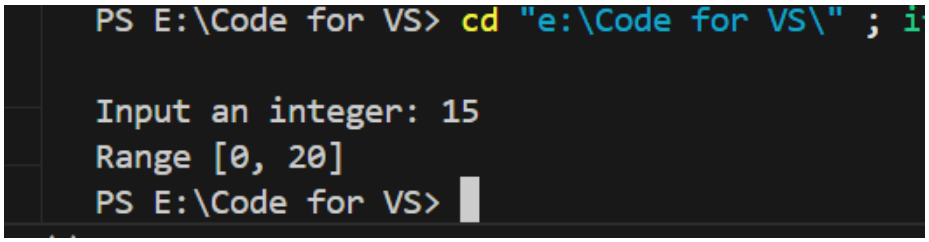
Problem 21:

Write a C program that reads an integer and checks the specified range to which it belongs. Print an error message if the number is negative and greater than 80.

Solution :

```
#include <stdio.h>
int main() {
   int x;
   printf("\nInput an integer: ");
   scanf("%d", &x);
   if(x >=0 && x <= 20)
   {
      printf("Range [0, 20]\n");
   }
   else if(x >=21 && x <= 40)
   {
      printf("Range (21,40]\n");
   }
   else if(x >=41 && x <= 60)
   {
      printf("Range (41,60]\n");
   }
   else if(x >61 && x <= 80)
   {printf("Range (61,80]\n");  }
   else
   {  printf("Outside the range\n");     }
   return 0;
}
```

Expected Output :



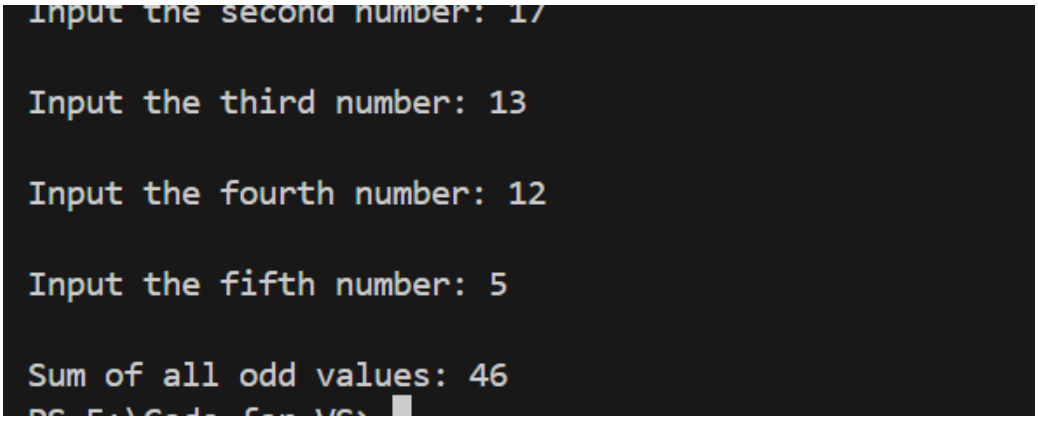
Problem 22:

Write a C program that reads 5 numbers and sums all odd values between them.

Solution :

```
#include <stdio.h>
int main() {
    int j, numbers[5], total=0;

    printf("\nInput the first number: ");
    scanf("%d", &numbers[0]);
    printf("\nInput the second number: ");
    scanf("%d", &numbers[1]);
    printf("\nInput the third number: ");
    scanf("%d", &numbers[2]);
    printf("\nInput the fourth number: ");
    scanf("%d", &numbers[3]);
    printf("\nInput the fifth number: ");
    scanf("%d", &numbers[4]);
    for(j = 0; j < 5; j++) {
       if((numbers[j]%2) != 0)
       {      total += numbers[j]; }
    }
    printf("\nSum of all odd values: %d", total);
    printf("\n");
    return 0;
}
```

Expected Output :



Problem 23:

Write a C program that reads three floating-point values and checks if it is possible to make a triangle with them. Determine the perimeter of the triangle if the given values are valid.

Solution :

```
#include <stdio.h>
int main() {
   float  x, y, z, P, A;

   printf("\nInput the first number: ");
   scanf("%f", &x);
   printf("\nInput the second number: ");
   scanf("%f", &y);
   printf("\nInput the third number: ");
   scanf("%f", &z);

   if(x < (y+z) && y < (x+z) && z < (y+x))
    {
      P = x+y+z;
      printf("\nPerimeter  = %.1f\n", P);
   }
   else
   {
      printf("Not possible to create a triangle..!");
   }
   return 0;
}
```

Expected Output :

```
Input the first number: 25

Input the second number: 15

Input the third number: 35

Perimeter  = 75.0
```

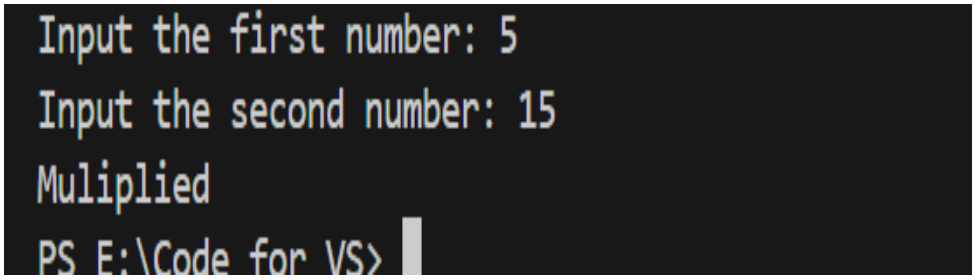
Problem 24:

Write a C program that reads two integers and checks whether they are multiplied or not.

Solution :

```
#include <stdio.h>
int main() {
   int x, y;
   printf("Input the first number: ");
   scanf("%d",&x);
   printf("Input the second number: ");
   scanf("%d",&y);
   if(x>y){
    int temp = x;
    x = y;
    y = temp;
   }
   if(y%x==0){
    printf("Muliplied\n");
   }
   else{
   printf("Not Multiplied!\n");
   }
    return 0;
}
```

Expected Output :



Problem 25:

Write a C program that reads an integer between 1 and 12 and prints the month of the year in English.

Solution :

```
#include <stdio.h>
int main() {
   int mno;
   printf("\nInput a number between 1 to 12 to get the month
name: ");
   scanf("%d", &mno);
   switch(mno) {
      case 1 : printf("January\n"); break;
      case 2 : printf("February\n"); break;
      case 3 : printf("March\n"); break;
      case 4 : printf("April\n"); break;
      case 5 : printf("May\n"); break;
      case 6 : printf("June\n"); break;
      case 7 : printf("July\n"); break;
      case 8 : printf("August\n"); break;
      case 9 : printf("September\n"); break;
      case 10 : printf("October\n"); break;
      case 11 : printf("November\n"); break;
      case 12 : printf("December\n"); break;
      default : printf("Input a number between 1 to 12.");
   }
return 0;
}
```

Expected Output :

```
Input a number between 1 to 12 to get the month name: 8
August
PS E:\Code for VS>
```

Problem 26:

Write a C program that prints all even numbers between 1 and 50 (inclusive).

Solution :

```
#include <stdio.h>
int main() {
    int n;
    scanf("%d",&n);

    for(int i=1; i<=50; i++){
      // scanf("%d",&i);
       if(i%2==0 && i>0){
           printf("%d ",i);
       }
    }
    return 0;
}
```

Expected Output :

```
50
2 4 6 8 10 12 14 16 18 20 22 24 26 28 30 32 34 36 38 40 42 44 46 48 50
PS F:\Code for VS>
```

Problem 27:

Write a C program that reads 5 numbers and counts the number of positive numbers and negative numbers.

Solution :

```
#include <stdio.h>
int main() {
   int n, m, o, p, q, count1 = 0, count2 = 0;
    printf("Input the first number: ");
    scanf("%d",&n);
     printf("Input the second number: ");
     scanf("%d",&m);
     printf("Input the third number: ");
     scanf("%d",&o);
     printf("Input the fourth number: ");
     scanf("%d",&p);
     printf("Input the fifth number: ");
     scanf("%d",&q);
       int numbers[] = {n, m, o, p, q};
    for (int i = 0; i < 5; i++) {
        if (numbers[i] > 0) {
            count1++;
        } else if (numbers[i] < 0) {
            count2++;
        }
    }  printf("Number of positive numbers: %d\n", count1);
    printf("Number of negative numbers: %d\n", count2);
    return 0;
}
```

Expected Output :

```
Input the first number: 5
Input the second number: 6
Input the third number: -3
Input the fourth number: 0
Input the fifth number: -6
Number of positive numbers: 2
Number of negative numbers: 2
```

Problem 28:

Write a C program that reads 5 numbers, counts the number of positive numbers, and prints out the average of all positive values.

Solution :

```
#include <stdio.h>

int main() {
    int n, m, o, p, q, count = 0, sum = 0;
    printf("Input the first number: ");scanf("%d", &n);
    printf("Input the second number: ");scanf("%d", &m);
    printf("Input the third number: ");scanf("%d", &o);
    printf("Input the fourth number: "); scanf("%d", &p);
    printf("Input the fifth number: ");  scanf("%d", &q);
    int numbers[] = {n, m, o, p, q};
    for (int i = 0; i < 5; i++) {
        if (numbers[i] > 0) {
            count++;
            sum += numbers[i];
        } }
    if (count > 0) {
        printf("Number of positive numbers: %d\n", count);
        printf("Average value of the said positive numbers: %.2f\n",
(float)sum / count);
    } else {
        printf("No positive numbers were entered.\n");
    }
    return 0;
}
```

Expected Output :

```
Input the first number: 5
Input the second number: 8
Input the third number: 10
Input the fourth number: -5
Input the fifth number: 25
Number of positive numbers: 4
Average value of the said positive numbers: 12.00
```

Problem 29:

Write a C program that read 5 numbers and sum of all odd values between them.

Solution :

```
#include <stdio.h>
int main() {
    int n, m, o, p, q;
    int sum = 0;
    printf("Input the first number: ");
    scanf("%d", &n);
    printf("Input the second number: ");
    scanf("%d", &m);
    printf("Input the third number: ");
    scanf("%d", &o);
    printf("Input the fourth number: ");
    scanf("%d", &p);
    printf("Input the fifth number: ");
    scanf("%d", &q);
    int numbers[] = {n, m, o, p, q};
    for (int i = 0; i < 5; i++) {
        if (numbers[i] % 2 != 0) {
            sum += numbers[i];       }
    }
      printf("Sum of all odd values: %d\n", sum);
    return 0;
}
```

Expected Output :

```
Input the first number: 5
Input the second number: 7
Input the third number: 9
Input the fourth number: 10
Input the fifth number: 13
Sum of all odd values: 34
```

Problem 30:

Write a C program to find and print the square of all the even values from 1 to a specified value.

Solution :

```
#include <stdio.h>
int main() {
    int x, i;
    printf("Input an integer: ");
    scanf("%d", &x);
    printf("List of square of each one of the even values from 1 to a
%d :\n", x);

    for(i = 2; i <= x; i++) {
        if((i%2) == 0) {
            printf("%d^2 = %d\n", i, i*i);
        }
    }

    return 0;
}
```

Expected Output :

```
Input an integer: 4
List of square of each one of the even values from 1 to a 4 :
2^2 = 4
4^2 = 16
```

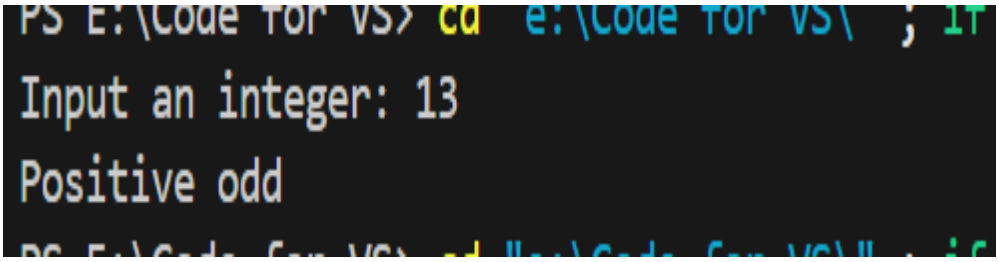
Problem 31:

Write a C program to check whether a given integer is positive even, negative even, positive odd or negative odd. Print even if the number is 0.

Solution :

```
#include <stdio.h>
int main() {
    int x;
    printf("Input an integer: ");
    scanf("%d",&x);
    if(x>0 && ((x%2)==0)){
        printf("Positive Even");
    }
      else if(x>0 && ((x%2)!=0)){
        printf("Positive odd");
    }
     else if(x<0 && ((x%2)!=0)){
        printf("Negative odd");
    }
   else{
     printf("Negative even");
   }
    return 0;
}
```

Expected Output :



Problem 32:

Write a C program to print all numbers between 1 and 100 which are divided by a specified number and the remainder will be 3.

Solution :

```
#include <stdio.h>
int main() {
   int x;
   printf("Input an integer: ");
   scanf("%d",&x);

   for(int i=1; i<=100; i++){

   if(i%x==3){
      printf("%d\n",i);
   }
   }

   return 0;
}
```

Expected Output :

```
Input an integer: 25
3
28
53
78
```

Problem 33:

Write a C program that accepts some integers from the user and finds the highest value and the input position.

Solution :

```
#include <stdio.h>
#define MAX 5
int main()
{
    int number[MAX], i, j, max=0, num_pos=0;
   printf("Input 5 integers: \n");

   for(i = 1; i <= MAX; i++) {
      scanf(" %d", &number[i]);
   }
   for(j = 1; j <= MAX; j++)
   {
      if(number[j] > max) {
         max = number[j];
         num_pos = j;
      }
   }
   printf("Highest value: %d\nPosition: %d\n", max, num_pos);
   return 0;
}
```

Expected Output :

```
Input 5 integers:
5
7
15
45
9
Highest value: 45
Position: 4
```

Problem 34:

Write a C program to compute the sum of consecutive odd numbers from a given pair of integers.

Solution :

```
#include <stdio.h>
int main() {
    int x, y, i, total = 0;
    printf("Input a pair of numbers (for example 10,2):\n");
    printf("Input first number of the pair: ");
    scanf("%d", &x);
    printf("Input second number of the pair: ");
    scanf("%d", &y);
    if (x < y) {
        return 0;
    }
    printf("List of odd numbers: \n");
    for (i = y; i <= x; i++) {
        if ((i % 2) != 0) {
            printf("%d\n", i);
            total+=i;
        }
    }
    printf("Sum = %d\n", total);
    return 0;
}
```

Expected Output :

```
Input first number of the pair: 10 2
Input second number of the pair: List of odd numbers:
3
5
7
9
Sum = 24
```

Problem 35:

Write a C program to check if two numbers in a pair are in ascending order or descending order.

Solution :

```
#include <stdio.h>
int main() {
   int x, y, i, total = 0;
   printf("Input a pair of numbers (for example 10,2 : 2,10):
\n");
   printf("Input first number of the pair: ");
   scanf("%d", &x);
   printf("Input second number of the pair: ");
   scanf("%d", &y);
   if (x > y) {
      printf("The pair is in descending order!");

   }
   else {
      printf("The pair is in ascending order!");
   }
   return 0;
}
```

Expected Output :

```
Input first number of the pair: 10
Input second number of the pair: 2
The pair is in descending order!
```

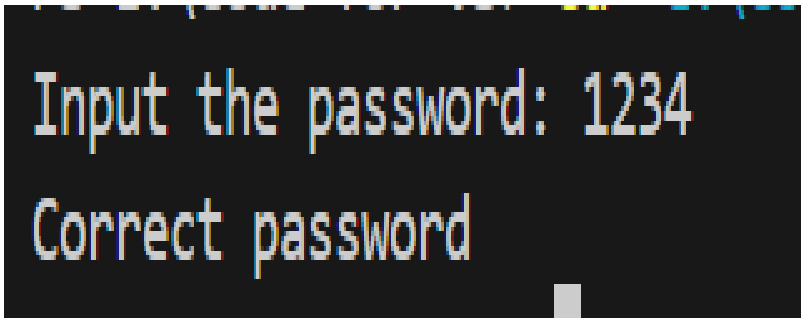
Problem 36:

Write a C program to read a password until it is valid. For wrong password print "Incorrect password" and for correct password print, "Correct password" and quit the program. The correct password is 1234.

Solution :

```
#include <stdio.h>
#define password 1234

int main() {
   int n;
   printf("Input the password: ");
   scanf("%d",&n);

   if(n==password){
    printf("Correct password\n");
   }
   else{
    printf("Incorrect pssword\n");
   }
    return 0;
}
```

Expected Output :



Problem 37:

Determine quadrant of Cartesian coordinates (x, y)

Solution :

```
#include <stdio.h>
int main() {
   int x, y;
   printf("Input the Coordinate(x,y):\n");
   printf("x: ");scanf("%d",&x);
   printf("y: ");scanf("%d",&y);
   if(x>0 && y>0){
    printf("Quadrant-I(+,+)\n");
   }
   if(x<0 && y>0){
    printf("Quadrant-II(-,+)\n");
   }
   if(x<0 && y<0){
    printf("Quadrant-III(-,-)\n");
   }
   if(x>0 && y<0){
    printf("Quadrant-IV(+,-)\n");
   }
return 0;
}
```

Expected Output :

```
Input the Coordinate(x,y):
x: 10
y: -3
Quadrant-IV(+,-)
```

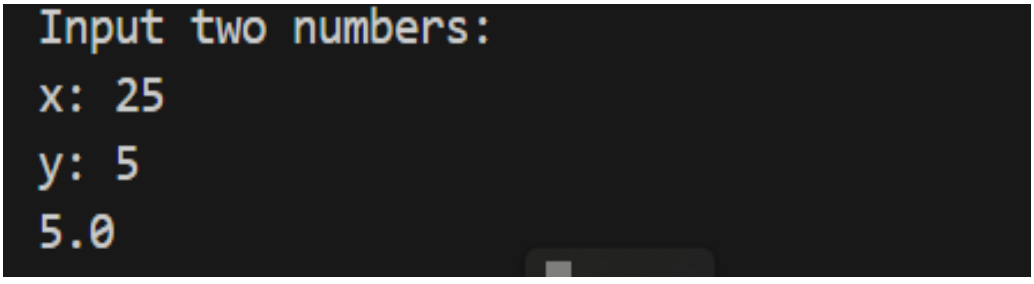
Problem 38:

Write a program that reads two numbers and divides the first number by the second number. If division is not possible print "Division is not possible".

Solution :

```
#include <stdio.h>
int main() {
    int x, y;
    float div_result;

    printf("Input two numbers: ");
    printf("\nx: ");
    scanf("%d", &x);
    printf("y: ");
    scanf("%d", &y);

    if(y != 0) {
        div_result = x/y;
        printf("%.1f\n", div_result);
    } else {
        printf("Division not possible.\n");
    }

    return 0;
}
```

Expected Output :



Problem 39:

Write a C program to calculate the sum of all numbers not divisible by 17 between two given integer numbers.

Solution :

```
#include <stdio.h>
int main() {
   int x, y, temp, i, sum=0;
   printf("\nInput the first integer: "); scanf("%d", &x);
   printf("\nInput the second integer: "); scanf("%d", &y);
  if(x > y) {
      temp = y;
      y = x;
      x = temp;
   }
   for(i = x; i <= y; i++) {
      if((i % 17) != 0) {
         sum += i;
      }
   }
   printf("\nSum: %d\n", sum);
   return 0;
}
```

Expected Output :

```
Input the first integer: 50

Input the second integer: 99

Sum: 3521
```

Problem 40:

Write a C program that finds all integer numbers that divide by 7 and have a remainder of 2 or 3 between two given integers.

Solution :

```
#include <stdio.h>
int main() {
   int x, y, temp, i, sum=0;
   printf("\nInput the first integer: ");
   scanf("%d", &x);
   printf("\nInput the second integer: ");
   scanf("%d", &y);
   if(x > y) {
      temp = y;
      y = x;
      x = temp;
   }
   for(i = x+1; i < y; i++) {
      if((i%7) == 2 || (i%7) == 3) {
         printf("%d\n", i);
      }
   }
   return 0;
}
```

Expected Output :

```
Input the first integer: 25
Input the second integer: 45
30
31
37
38
44
```

Problem 41:

Write a C program to print 3 numbers on a line, starting with 1 and printing n lines. Accept the number of lines (n, integer) from the user.

Solution :

```
#include <stdio.h>
int main() {
   int a, i, j = 1, x = 0;

   printf("Input number of lines: ");
   scanf("%d", &a);

   for(i = 1; i <= a; i++) {
      while(x < 3) {
         printf("%d ", j++);
         x++;
      }
      x = 0;
      printf("\n");
   }

   return 0;
}
```

Expected Output :

```
Input number of lines: 5
1 2 3
4 5 6
7 8 9
10 11 12
13 14 15
```

Problem 42:

Write a C program to print a number, its square and cube, starting with 1 and printing n lines. Accept the number of lines (n, integer) from the user.

Solution :

```
#include <stdio.h>
int main() {
   int a, i, j = 1, x = 0;

   printf("Input number of lines: ");
   scanf("%d", &a);

   for(i = 1; i <= a; i++) {

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

   return 0;
}
```

Expected Output :

```
Input number of lines: 5
1 1 1
2 4 8
3 9 27
4 16 64
5 25 125
```

Problem 43:

Write a C program that reads two integers p and q, prints p number of lines in a sequence of 1 to b in a line.

Solution :

```
#include <stdio.h>
int main() {
   int x, y, i, j, l;

   printf("Input number of lines: ");
   scanf("%d", &x);
   printf("Number of numbers in a line: ");
   scanf("%d", &y);

   for(i = 1, l=1; i <= x; i++) {
      for(j = 1; j <= y; j++) {
         printf("%d ", l);
         l++;
      }
      printf("\n");
   }

   return 0;
}
```

Expected Output :

```
Input number of lines: 5
Number of numbers in a line: 6
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
```

Problem 44:

Write a C program to calculate the average mathematics marks of some students. Input 0 (excluding to calculate the average) or a negative value to terminate the input process.

Solution :

```
#include <stdio.h>
int main() {
    int marks[99], m, i, a=0, total=0;
    float f;
    printf("Input Mathematics marks (0 to terminate): ");
    for(i = 0; ; i++) {
        scanf("%d", &marks[i]);
        if(marks[i] <= 0) {
            break;
        }
        a++;
        total += marks[i];
    }
    f = (float)total/(float)a;
    printf("Average marks in Mathematics: %.2f\n", f);
    return 0;
}
```

Expected Output :

```
Input Mathematics marks (0 to terminate): 15
20
25
0
Average marks in Mathematics: 20.00
```

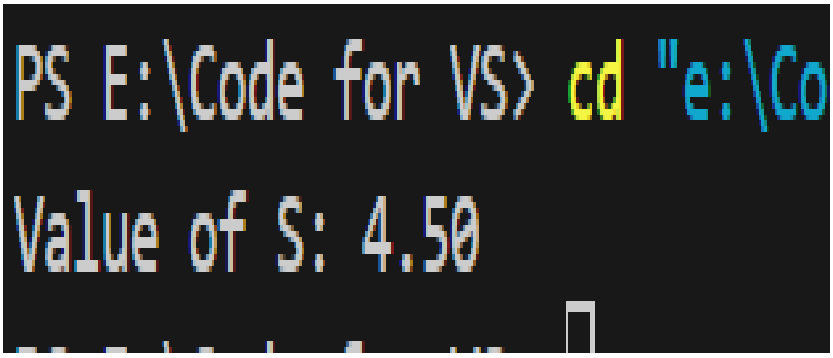
Problem 45:

Write a C program to calculate the value of S where S = 1 + 1/2 + 1/3 + … + 1/50.

Solution :

```
#include <stdio.h>
int main() {
    float S = 0;
    int i;

    for(i=1; i<=50; i++) {
        S += (float)1/i;
    }

    printf("Value of S: %.2f\n", S);

    return 0;
}
```

Expected Output :



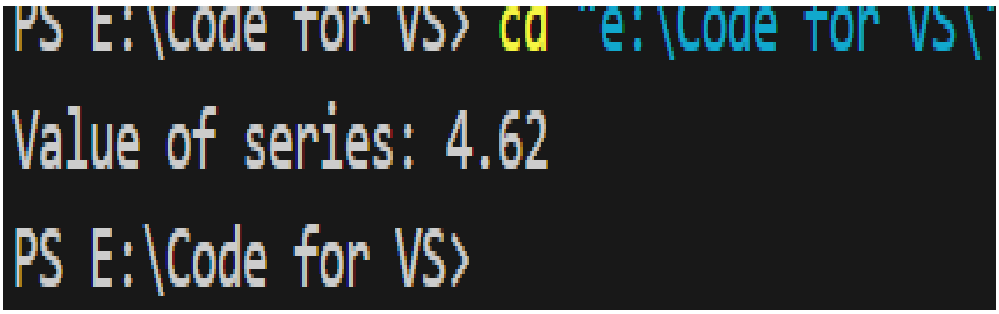
Problem 46:

Write a C program to calculate the value of S where S = 1 + 3/2 + 5/4 + 7/8.

Solution :

```
#include <stdio.h>
int main() {
    double s=0, j=1, d, i;

    for(i=1; i<=7; i+=2){
        d = (i/j);
        s += d;
        j = j*2;
    }
    printf("Value of series: %.2lf\n", s);

    return 0;
}
```

Expected Output :



Problem 47:

Write a C program that finds all the divisors of an integer.

Solution :

```
#include <stdio.h>

int main() {
   int n,count=0;

   printf("Input an integer: ");
   scanf("%d",&n);

  for(int i=1; i<=n; i++){
   if(n%i==0){
    printf("%d\n",i);
   }
}
    return 0;
}
```

Expected Output :

```
Input an integer: 45
1
3
5
9
15
45
```

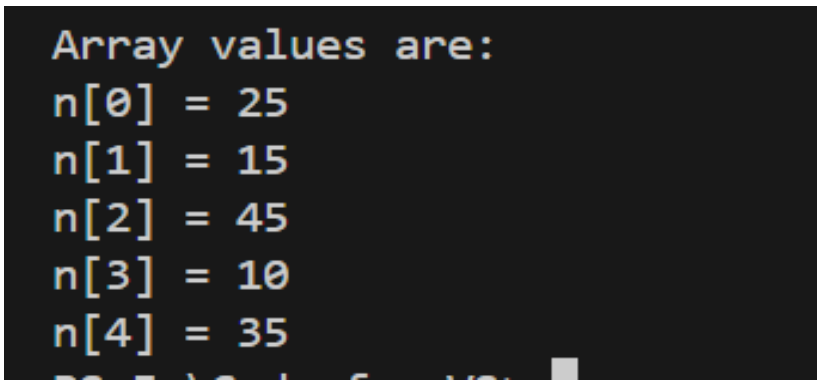
Problem 48:

Write a C program that reads and prints the elements of an array of length 7. Before printing, replace every negative number, zero, with 100.

Solution :

```
#include <stdio.h>
int main() {
   int n[5], i, x;
   printf("Input the 5 members of the array:\n");
   for(i = 0; i < 5; i++) {
      scanf("%d", &x);
      if(x > 0) {
         n[i] = x;
      } else {
         n[i] = 100;
      }
   }
   printf("Array values are: \n");
   for(i = 0; i < 5; i++) {
      printf("n[%d] = %d\n", i, n[i]);
   }
      return 0;
}
```

Expected Output :



Problem 49:

Write a C program to read and print the elements of an array with length 7. Before printing, insert the triple of the previous position, starting from the second position.

Solution :

```
#include <stdio.h>
int main() {
   int n[5], i, x;

   printf("Input the first number of the array:\n");
   scanf("%d", &x);
   for(i = 0; i < 5; i++) {
      n[i] = x;
      x = 3 * x;
   }
   for(i = 0; i < 5; i++) {
      printf("n[%d] = %d\n", i, n[i]);
   }
   return 0;
}
```

Expected Output :

```
Input the first number of the array:
5
n[0] = 5
n[1] = 15
n[2] = 45
n[3] = 135
n[4] = 405
```

Problem 50:

Write a C program to read an array of length 5 and print the position and value of the array elements of value less than 5.

Solution :

```
#include <stdio.h>
#define AL 5
#define MAX 5

int main() {
   float N[AL];
   int i;
   printf("Input the 5 members of the array:\n");
   for(i = 0; i < AL; i++) {
      scanf("%f", &N[i]);
   }
   for(i = 0; i < AL; i++) {
      if(N[i] < MAX) {
         printf("A[%d] = %.1f\n", i, N[i]);
      }
   }
   return 0;
}
```

Expected Output :

```
Input the 5 members of the array:
15
25
4
35
40
A[2] = 4.0
```