1. Write a C program to print your name, date of birth. and mobile number.
Expected Output:
Name : Alexandra Abramov
DOB : July 14, 1975
Mobile: 99-9999999999

2. Write a C program to print a block F using hash (#), where the F has a height of six characters and width of five and four characters.
Expected Output:
```
######
#
#
#####
#
#
#
```

3. Write a C program to print a big 'C'.
Expected Output:
```
  ######
 ##    ##
#
#
#
#
 ##    ##
  ######
```

4. Write a C program to print the following characters in a reverse way.
Test Characters: 'X', 'M', 'L'
Expected Output:
The reverse of XML is LMX

5. Write a C program to compute the perimeter and area of a rectangle with a height of 7 inches. and width of 5 inches.
Expected Output:
Perimeter of the rectangle = 24 inches
Area of the rectangle = 35 square inches

6. Write a C program to compute the perimeter and area of a circle with a radius of 6 inches.
Expected Output:
Perimeter of the Circle = 37.680000 inches
Area of the Circle = 113.040001 square inches
7. 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
Declaration :

```
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;
```

8. Write a C program to convert specified days into years, weeks and days.
Note: Ignore leap year.
Test Data:
Number of days: 1329
Expected Output:
Years: 3
Weeks: 33
Days: 3

9. Write a C program that accepts two integers from the user and calculate the sum of the two integers.
Test Data:
Input the first integer: 25
Input the second integer: 38
Expected Output:
Sum of the above two integers = 63

10. Write a C program that accepts two integers from the user and calculate the product of the two integers.
Test Data:
Input the first integer: 25
Input the second integer: 15
Expected Output:
Product of the above two integers = 375

11. Write a C program that accepts two item's weight (floating points' values) and number of purchase (floating points' values) and calculate the average value of the items.
Test Data:
Weight - Item1: 15
No. of item1: 5
Weight - Item2: 25
No. of item2: 4
Expected Output:
Average Value = 19.444444

12. Write a C program that accepts an employee's ID, total worked hours of a month and the amount he received per hour. Print the employee's ID and salary (with two decimal places) of a particular month.
Test Data:
Input the Employees ID(Max. 10 chars): 0342
Input the working hrs: 8
Salary amount/hr: 15000
Expected Output:

Employees ID = 0342
Salary = U$ 120000.00

13. Write a C program that accepts three integers and find the maximum of three.
Test Data:
Input the first integer: 25
Input the second integer: 35
Input the third integer: 15
Expected Output:
Maximum value of three integers: 35

14. Write a C program to calculate a bike's average consumption from the given total distance (integer value) traveled (in km) and spent fuel (in liters, float number – 2 decimal point).
Test Data:
Input total distance in km: 350
Input total fuel spent in liters: 5
Expected Output:
Average consumption (km/lt) 70.000

15. Write a C program to calculate the distance between the two points.
Test Data:
Input x1: 25
Input y1: 15
Input x2: 35
Input y2: 10
Expected Output:
Distance between the said points: 11.1803

16. Write a C program to read an amount (integer value) and break the amount into smallest possible number of bank notes.
Test Data:
Input the amount: 375
Expected Output:

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

17. Write a C program to convert a given integer (in seconds) to hours, minutes and seconds.
Test Data:
Input seconds: 25300
Expected Output:
There are:
H:M:S - 7:1:40

18. Write a C program to convert a given integer (in days) to years, months and days, assumes that all months have 30 days and all years have 365 days.
Test Data:
Input no. of days: 2535
Expected Output:
6 Year(s)
11 Month(s)
15 Day(s)

19. Write a C program that accepts 4 integers p, q, r, s from the user where 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".
Test Data:
Input the second integer: 35
Input the third integer: 15
Input the fourth integer: 46

Expected Output:
Wrong values

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.
Test Data:
Input the first number(a): 25
Input the second number(b): 35
Input the third number(c): 12
Expected Output:
Root1 = -0.60000
Root2 = -0.80000

21. Write a C program that reads an integer and check the specified range where it belongs. Print an error message if the number is negative and greater than 80.
Test Data:
Input an integer: 15
Expected Output:
Range [0, 20]

22. Write a C program that read 5 numbers and sum of all odd values between them.
Test Data:
Input the first number: 11
Input the second number: 17
Input the third number: 13
Input the fourth number: 12
Input the fifth number: 5
Expected Output:
Sum of all odd values: 46

22. Write a C program that reads three floating values and check if it is possible to make a triangle with them. Also calculate the perimeter of

the triangle if the said values are valid.
Test Data:
Input the first integer: 5
Input the second integer: 15
Input the third integer: 10

Expected Output:
Numbers in sorted order: 5 10 15

23. Write a C program that reads three floating values and check if it is possible to make a triangle with them. Also calculate the perimeter of the triangle if the said values are valid.
Test Data:
Input the first number: 25
Input the second number: 15
Input the third number: 35
Expected Output:
Perimeter = 75.0

24. Write a C program that reads two integers and checks if they are multiplied or not.
Test Data:
Input the first number: 5
Input the second number: 15
Expected Output:
Multiplied!

25. Write a C program that reads an integer between 1 and 12 and print the month of the year in English.
Test Data:
Input a number between 1 to 12 to get the month name: 8
Expected Output:
August

26. Write a C program that prints all even numbers between 1 and 50 (inclusive).
Test Data:
Even numbers between 1 to 50 (inclusive):
Expected Output:
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

27. Write a C program that read 5 numbers and counts the number of positive numbers and negative numbers.
Test Data:
Input the first number: 5
Input the second number: -4
Input the third number: 10
Input the fourth number: 15
Input the fifth number: -1
Expected Output:
Number of positive numbers: 3
Number of negative numbers: 2

28. Write a C program that read 5 numbers and counts the number of positive numbers and print the average of all positive values.
Test Data:
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
Expected Output:
Number of positive numbers: 4
Average value of the said positive numbers: 12.00

29. Write a C program that read 5 numbers and sum of all odd values between them.
Test Data:
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
Expected Output:
Sum of all odd values: 34

30. Write a C program to find and print the square of each one of the even values from 1 to a specified value.
Test Data:
List of square of each one of the even values from 1 to a 4 :
Expected Output:
2^2 = 4
4^2 = 16

31. Write a C program to check a given integer is positive even, negative even, positive odd or negative odd. Print even if the number is 0.
Test Data:
Input an integer: 13
Expected Output:
Positive Odd

32. Write a C program to print all numbers between 1 to 100 which divided by a specified number and the remainder will be 3.
Test Data:
Input an integer: 25
Expected Output:
3
28
53
78
103
128
153
178

203
228
253
278
303
328
353
378
403
428
453
478

33. Write a C program that accepts some integers from the user and find the highest value and the input position.
Test Data:
Input 5 integers:
5
7
15
23
45
Expected Output:
Highest value: 45
Position: 5

34. Write a C program to print the numbers from the lowest to the highest (inclusive) and the sum of consecutive integers from a given pair of numbers.
Test Data:
Input a pair of numbers (for example 10,2):
Input first number of the pair: 10
Input second number of the pair: 2
Expected Output:
List of odd numbers: 3

5
7
9
Sum=24

35. Write a C program to check if two numbers in a pair is in ascending order or descending order.
Test Data:
Input a pair of numbers (for example 10,2 : 2,10):
Input first number of the pair: 10
Expected Output:
Input second number of the pair: 2
The pair is in descending order!

36. Write a C program to read a password until it is correct. For wrong password print "Incorrect password" and for correct password print "Correct password" and quit the program. The correct password is 1234.
Test Data:
Input the password: 1234
Expected Output:
Correct password

37. Write a C program to read the coordinates(x, y) (in Cartesian system) and find the quadrant to which it belongs (Quadrant -I, Quadrant -II, Quadrant -III, Quadrant -IV).
Note: A Cartesian coordinate system is a coordinate system that specifies each point uniquely in a plane by a pair of numerical coordinates.
These are often numbered from 1st to 4th and denoted by Roman numerals: I (where the signs of the (x,y) coordinates are I(+,+), II (−,+), III (−,−), and IV (+,−).
Test Data:
Input the Coordinate(x,y):
x: 25
y: 15

Expected Output:
Quadrant-I(+,+)

38. Write a program that reads two numbers and divide the first number by second number. If the division not possible print "Division not possible".
Test Data:
Input two numbers:
x: 25
y: 5
Expected Output: 5.0

39. Write a C program to calculate the sum of all number divisible by 17 between two given integer numbers.
Test Data:
Input the first integer: 50 Input the second integer: 99
Expected Output:
Sum: 3521

40. Write a C program to find all numbers which dividing it by 7 and the remainder is equal to 2 or 3 between two given integer numbers.
Test Data:
Input the first integer: 25
Input the second integer: 45
Expected Output:
30
31
37
38
44

41. Write a C program to print 3 numbers in a line, starting from 1 and print n lines. Accept number of lines (n, integer) from the user.
Test Data:
Input number of lines: 5

Expected Output:
1 2 3
4 5 6
7 8 9
10 11 12
13 14 15

42. Write a C program to print a number, it's square and cube in a line, starting from 1 and print n lines. Accept number of lines (n, integer) from the user.
Test Data:
Input number of lines: 5
Expected Output:
1 1 1
2 4 8
3 9 27
4 16 64
5 25 125

43. Write a C program that reads two integers p and q, print p number of lines in a sequence of 1 to b in a line.
Test Data:
Input number of lines: 5
Number of characters in a line: 6
Expected Output:
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

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

Input Mathematics marks (0 to terminate): 10
15
20
25
0
Expected Output:
Average marks in Mathematics: 17.50

45. Write a C program to calculate the value of S where S = 1 + 1/2 + 1/3 + … + 1/50.
Expected Output:
Value of S: 4.50

46. Write a C program to calculate the value of S where S = 1 + 3/2 + 5/4 + 7/8.
Expected Output:
Value of series: 4.62

47. Write a C program that reads an integer and find all its divisor.
Test Data:
Input an integer: 45
Expected Output:
All the divisor of 45 are:
1
3
5
9
15
45

48. Write a C program to read and print the elements of an array of length 7, before print replace every negative number, zero with 100.
Test Data:
Input the 5 members of the array:
25

45
35
65
15

Expected Output:
Array values are:
n[0] = 25
n[1] = 45
n[2] = 35
n[3] = 65
n[4] = 15

49. Write a C program to read and print the elements of an array of length 7, before print, put the triple of the previous position starting from the second position of the array.
For example, if the first number is 2, the array numbers must be 2, 6, 18, 54 and 162
Test Data:
Input the first number of the array: 5
Expected Output:
n[0] = 5
n[1] = 15
n[2] = 45
n[3] = 135
n[4] = 405

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.
Test Data:
Input the 5 members of the array:
15
25
4
35

40
Expected Output:
A[2] = 4.0

51. Write a C program to read an array of length 6, change the first element by the last, the second element by the fifth and the third element by the fourth. Print the elements of the modified array.
Test Data:
Input the 5 members of the array:
15
20
25
30
35

Expected Output:
array_n[0] = 35
array_n[1] = 30
array_n[2] = 25
array_n[3] = 20
array_n[4] = 15

52. Write a C program to read an array of length 6 and find the smallest element and its position.
Test Data:
Input the length of the array: 5 Input the array elements:
25
35
20
14
45
Expected Output:
Smallest Value: 14
Position of the element: 3