1. Write a C program to print all the composite numbers (that have factors other than 1 and the number itself) from 2 to n. NB- n will be any integer number taken by user.	**Input** Enter number: 15 **Output** composite numbers are: 4, 6, 8, 9, 10, 12, 14, 15
2. Write a C program to take two inputs as number of line and limit of numbers printing per line. Print the odd number pattern when the line number is odd starting from the line number itself and print the even number pattern when the line number is even and the pattern starts from the line number itself. Output should be same as shown.	**Input** Enter the number of line: 4 Enter the limit of numbers per line: 5 **Output** Odd Line (1): 1,3,5,7,9 Even Line (2): 2,4,6,8,10 Odd Line (3): 3,5,7,9,11 Even Line (4): 4,6,8,10,12
3. Write a C program to print the leap years between n and m, where n and m are two inputs taken by user as years. Also count the number of leap years between them. N.B: If no leap year exist, then show the message " No leap year found"	**Input** Enter starting year = 2000 Enter ending year = 2018 **Output** Leap years between 2000 to 2018 are: 2000, 2004, 2008, 2012, 2016 The number of leap year is 5
4. Write a C program to find all the prime numbers (that is divisible by only 1 and itself) between 1 to n. NB- n will be any integer number taken by user.	**Input** Enter number : 15 **Output** Prime numbers between 1 to 15 are: 1, 2, 3, 5, 7, 11, 13
5. Write a C program to find factorial of every number between 1 to N. N.B: Factorial means the product of all the whole numbers from 1 to N. 4! = 1*2*3*4 = 24	**Input** Enter number: 5 **Output** Factorial of 1: 1 Factorial of 2: 2 Factorial of 3: 6 Factorial of 4: 24 Factorial of 5: 120

6. Write a C program to print the factorial number of every digit of an imputed number and determine whether it is a prime number or not.	**Input** Enter a number: 379 **Output** 9! = 362880 9 is not a prime number 7! = 5040 7 is a prime number 3! = 6 3 is a prime number
7. Write a C program to print the number pattern	**Input** Enter the line limit: 5 **Output** 1 1 1 1 1 2 2 2 2 3 3 3 4 4 5
8. Write a C program to print the number pattern	**Input** Enter the number: 7 **Output** 1 1 0 1 0 1 1 0 1 0 1 0 1 0 1 1 0 1 0 1 0 1 0 1 0 1 0 1
9. Write a C program to print the Star (*) pattern.	**Input** Enter the number: 5 **Output** * *** ***** ******* *********

10. Write a C program to print the Star (*) pattern.	**Input** Enter the number: 5 **Output** * *** ***** ******* ********* ******* ***** *** *
11. Write a C program to print the following pattern.	**Input** Enter the number: 7 **Output** 1234567 234567 34567 4567 567 67 7
12. Write a C program to print the following pattern.	**Input** Enter the number: 7 **Output** 1234567 234567 34567 4567 567 67 7 67 567 4567 34567 234567 1234567

13. Write a C program to print the following pattern.	**Input** 7 **Output** 0000000 0100000 0020000 0003000 0000400 0000050 0000006
14. Write a C program to print the following pattern. N.B: In the output the value in the following pattern increases vertically (Up to bottom).	**Input** 5 **Output** 1 2 6 3 7 10 4 8 11 13 5 9 12 14 15
15. Write a C program to take several inputs such as, StuNum = number of student of a batch. CourseNum= Number of courses taken by them. (Assuming that they have taken same number of courses.) mid1= mid 1 number for an individual course and for an individual student. mid2= mid 2 number for an individual course and for an individual student. final= Final number for an individual course and for an individual student. [Suppose there is three exam as mid1, mid2 and final for every course] Now calculate 1. Their individual grade for each courses. 2. Their individual average grade of all courses. 3. Average grade of all students of that batch.	**Input** Enter the number of student: 20 Enter how many courses they have taken: 2 Enter the mid 1 number for course 1 for student no 1 = 30 Enter the mid 2 number for course 1 for student no 1 = 25 Enter the final number for course 1 for student no 1 = 40 {Here the following line will be output} Student no 1 get 4.00 in course 1. {Input continues} Enter the mid 1 number for course 2 for student no 1 = 40 Enter the mid 2 number for course 2 for student no 1 = 30 Enter the final number for course 2 for student no 1 = 27 {Here again the following line will be output} Student no 1 get 4.00 in course no 2. Student no 1 get average GPA 4.00 ----------

N.B: Use if-else to determine the grade. N.B: While testing your code, it is advised to take small number for StuNum<5 otherwise it would take much time just to check your full correct output. N.B: { } bracket use for making the question more understandable so these are not meant for printing. But print ---- sign after each students gpa print.	{Input continues} Enter the mid 1 number for course 2 for student no 2 = 40 Enter the mid 2 number for course 2 for student no 2 = 40 Enter the final number for course 2 for student no 2 = 7 {Here the following line will be output} Student no 2 get 3.70 in course 1. {Input continues} Enter the mid 1 number for course 2 for student no 2 = 40 Enter the mid 2 number for course 2 for student no 2 = 30 Enter the final number for course 2 for student no 2 = 18 {Here again the following line will be output} Student no 2 get 3.70 in course no 2. Student no 2 get average GPA 3.70 ---------- ////DO THIS PROCESS FOR StuNUM(20) time. ////ALL AT LAST PRINT THE AVERAGE GPA OF (1-StuNum) STUDENTS. {output} The total gpa average of 20 student is 3.56