

Amity School of Engineering &

Technology Noida (U.P.) 201303

ASSIGNMENT-2

Computer Science: Programming in C

Under the guidance of

DR. Shipra Shukla

Name: SHUBHAM THUKRAN Enrolment No.: A2305220734

Section: 1CSE14

Index

1. [Write a C program to print all natural numbers from 1 to n.](https://codeforwin.org/2015/06/c-program-to-print-natural-numbers-from-1-to-n.html)
2. [Write a C program to print all natural numbers in reverse (from n to 1).](https://codeforwin.org/2015/07/c-program-to-print-all-natural-numbers-in-reverse.html)
3. [Write a C program to print all alphabets from a to z.](https://codeforwin.org/2015/06/c-program-to-print-a-to-z.html)
4. [Write a C program to print all even numbers between 1 to 100.](https://codeforwin.org/2015/06/c-program-to-print-all-even-numbers-between-1-to-100.html)
5. [Write a C program to print all odd number between 1 to 100.](https://codeforwin.org/2015/06/c-program-to-print-all-odd--numbers-between-1-to-100.html)
6. [Write a C program to find sum of all natural numbers between 1 to n.](https://codeforwin.org/2015/06/c-program-to-calculate-sum-of-first-n-natural-numbers.html)
7. [Write a C program to find sum of all even numbers between 1 to n.](https://codeforwin.org/2015/06/c-program-to-print-sum-of-all-even-numbers-between-1-to-n.html)
8. [Write a C program to find sum of all odd numbers between 1 to n.](https://codeforwin.org/2015/06/c-program-to-print-sum-of-all-odd-numbers-between-1-to-n.html)
9. [Write a C program to print multiplication table of any number.](https://codeforwin.org/2015/06/c-program-to-print-table-of-any-number.html)
10. [Write a C program to count number of digits in a number.](https://codeforwin.org/2016/10/c-program-to-count-number-of-digits-in-number.html)
11. [Write a C program to find first and last digit of a number.](https://codeforwin.org/2015/06/how-to-find-first-and-last-digit-of-any-number.html)
12. [Write a C program to find sum of first and last digit of a number.](https://codeforwin.org/2015/06/c-program-to-find-sum-of-first-and-last-digit-of-number.html)
13. [Write a C program to swap first and last digits of a number.](https://codeforwin.org/2016/01/c-program-to-swap-first-and-last-digit-of-number.html)
14. [Write a C program to calculate sum of digits of a number.](https://codeforwin.org/2015/06/c-program-to-calculate-sum-of-digits.html)
15. [Write a C program to calculate product of digits of a number](https://codeforwin.org/2015/06/c-program-to-calculate-product-of-digits.html).
16. [Write a C program to enter a number and print its reverse](https://codeforwin.org/2015/06/c-program-to-find-reverse-of-any-number.html).
17. [Write a C program to check whether a number is palindrome or not.](https://codeforwin.org/2015/06/c-program-to-check-whether-number-is-palindrome-or-not.html)
18. [Write a C program to find frequency of each digit in a given integer.](https://codeforwin.org/2016/10/c-program-to-count-frequency-of-digits-in-number.html)
19. [Write a C program to enter a number and print it in words.](https://codeforwin.org/2015/06/c-program-to-print-numbers-in-words.html)
20. [Write a C program to print all ASCII character with their values](https://codeforwin.org/2015/06/c-program-to-print-ascii-values-of-all-characters.html).
21. [Write a C program to find power of a number using for loop](https://codeforwin.org/2015/07/c-program-to-find-power-of-number-using-for-loop.html).
22. [Write a C program to find all factors of a number.](https://codeforwin.org/2015/06/c-program-to-print-factors-of-any-number.html)
23. [Write a C program to calculate factorial of a number.](https://codeforwin.org/2015/06/c-program-to-calculate-factorial-of-any-number.html)
24. [Write a C program to find HCF (GCD) of two numbers](https://codeforwin.org/2015/06/c-program-to-find-hcf-of-two-numbers.html).
25. [Write a C program to find LCM of two numbers.](https://codeforwin.org/2015/06/c-program-to-find-lcm-of-two-numbers.html)
26. [Write a C program to check whether a number is Prime number or not.](https://codeforwin.org/2015/06/c-program-to-check-prime-number.html)
27. [Write a C program to print all Prime numbers between 1 to n.](https://codeforwin.org/2015/06/c-program-to-print-all-prime-numbers-between-1-to-n.html)
28. [Write a C program to find sum of all prime numbers between 1 to n.](https://codeforwin.org/2015/06/c-program-to-find-sum-of-all-prime.html)
29. [Write a C program to find all prime factors of a number](https://codeforwin.org/2015/06/c-program-to-find-all-prime-factors-of-any-number.html).
30. [Write a C program to check whether a number is Armstrong number or not.](https://codeforwin.org/2015/06/c-program-to-check-armstrong-number.html)
31. [Write a C program to print all Armstrong numbers between 1 to n.](https://codeforwin.org/2015/06/c-program-to-print-first-n-armstrong-numbers.html)
32. [Write a C program to check whether a number is Perfect number or not.](https://codeforwin.org/2015/06/c-program-to-check-perfect-number.html)
33. [Write a C program to print all Perfect numbers between 1 to n.](https://codeforwin.org/2015/06/c-program-to-print-all-perfect-numbers-between-1-to-n.html)
34. [Write a C program to check whether a number is Strong number or not.](https://codeforwin.org/2015/06/c-program-to-check-strong-number.html)
35. [Write a C program to print all Strong numbers between 1 to n.](https://codeforwin.org/2015/06/c-program-to-print-all-strong-numbers.html)
36. [Write a C program to print Fibonacci series up to n terms.](https://codeforwin.org/2015/06/fibonacci-series-in-c-program.html)
37. [Write a C program to find one's complement of a binary number.](https://codeforwin.org/2015/08/c-program-to-find-ones-complement-of-binary-number.html)
38. [Write a C program to find two's complement of a binary number.](https://codeforwin.org/2015/08/c-program-to-find-twos-complement-of-binary-number.html)
39. [Write a C program to convert Binary to Octal number system](https://codeforwin.org/2015/08/c-program-to-convert-binary-to-octal-number-system.html).
40. [Write a C program to convert Binary to Decimal number system.](https://codeforwin.org/2015/08/c-program-to-convert-binary-to-decimal-number-system.html)
41. [Write a C program to convert Binary to Hexadecimal number system.](https://codeforwin.org/2015/08/c-program-to-convert-binary-to-hexadecimal-number-system.html)
42. [Write a C program to convert Octal to Binary number system](https://codeforwin.org/2015/08/c-program-to-convert-octal-to-binary-number-system.html).
43. [Write a C program to convert Octal to Decimal number system](https://codeforwin.org/2015/08/c-program-to-convert-octal-to-decimal-number-system.html).
44. [Write a C program to convert Octal to Hexadecimal number system.](https://codeforwin.org/2015/08/c-program-to-convert-octal-to-hexadecimal-number-system.html)
45. [Write a C program to convert Decimal to Binary number system.](https://codeforwin.org/2015/08/c-program-to-convert-from-decimal-to-binary-number-system.html)
46. [Write a C program to convert Decimal to Octal number system](https://codeforwin.org/2015/08/c-program-to-convert-decimal-to-octal-number-system.html).
47. [Write a C program to convert Decimal to Hexadecimal number system](https://codeforwin.org/2015/08/c-program-to-convert-decimal-to-hexadecimal-number-system.html).
48. [Write a C program to convert Hexadecimal to Binary number system](https://codeforwin.org/2015/08/c-program-to-convert-hexadecimal-to-binary-number-system.html).
49. [Write a C program to convert Hexadecimal to Octal number system.](https://codeforwin.org/2015/09/c-program-to-convert-hexadecimal-to-octal-number-system.html)
50. [Write a C program to convert Hexadecimal to Decimal number system](https://codeforwin.org/2015/09/c-program-to-convert-hexadecimal-to-decimal-number-system.html).

# Pattern Exercises

1. [Star pattern programs - Write a C program to print the given star patterns.](https://codeforwin.org/2015/07/star-patterns-program-in-c.html)

[\*](https://codeforwin.org/2015/07/equilateral-triangle-star-pattern-program-in-c.html)

[\*\*\*](https://codeforwin.org/2015/07/equilateral-triangle-star-pattern-program-in-c.html)

[\*\*\*\*\*](https://codeforwin.org/2015/07/equilateral-triangle-star-pattern-program-in-c.html)

[\*\*\*\*\*\*\*](https://codeforwin.org/2015/07/equilateral-triangle-star-pattern-program-in-c.html)

[\*\*\*\*\*\*\*\*\*](https://codeforwin.org/2015/07/equilateral-triangle-star-pattern-program-in-c.html)

[Pyramid Star Pattern](https://codeforwin.org/2015/07/equilateral-triangle-star-pattern-program-in-c.html)

[\*](https://codeforwin.org/2015/07/c-program-to-print-hollow-pyramid-star-pattern.html)

[\* \*](https://codeforwin.org/2015/07/c-program-to-print-hollow-pyramid-star-pattern.html)

[\* \*](https://codeforwin.org/2015/07/c-program-to-print-hollow-pyramid-star-pattern.html)

[\* \*](https://codeforwin.org/2015/07/c-program-to-print-hollow-pyramid-star-pattern.html)

[\*\*\*\*\*\*\*\*\*](https://codeforwin.org/2015/07/c-program-to-print-hollow-pyramid-star-pattern.html)

[Hollow Pyramid Star Pattern](https://codeforwin.org/2015/07/c-program-to-print-hollow-pyramid-star-pattern.html)

[\*\*\*\*\*\*\*\*\*](https://codeforwin.org/2015/07/reverse-pyramid-star-pattern-in-c.html)

[\*\*\*\*\*\*\*](https://codeforwin.org/2015/07/reverse-pyramid-star-pattern-in-c.html)

[\*\*\*\*\*](https://codeforwin.org/2015/07/reverse-pyramid-star-pattern-in-c.html)

[\*\*\*](https://codeforwin.org/2015/07/reverse-pyramid-star-pattern-in-c.html)

[\*](https://codeforwin.org/2015/07/reverse-pyramid-star-pattern-in-c.html)

[Inverted Pyramid Star Pattern](https://codeforwin.org/2015/07/reverse-pyramid-star-pattern-in-c.html)

[\*\*\*\*\*\*\*\*\*](https://codeforwin.org/2015/07/c-program-to-print-hollow-inverted-pyramid-star-pattern.html)

[\* \*](https://codeforwin.org/2015/07/c-program-to-print-hollow-inverted-pyramid-star-pattern.html)

[\* \*](https://codeforwin.org/2015/07/c-program-to-print-hollow-inverted-pyramid-star-pattern.html)

[\* \*](https://codeforwin.org/2015/07/c-program-to-print-hollow-inverted-pyramid-star-pattern.html)

[\*](https://codeforwin.org/2015/07/c-program-to-print-hollow-inverted-pyramid-star-pattern.html)

[Hollow Inverted Pyramid Star Pattern](https://codeforwin.org/2015/07/c-program-to-print-hollow-inverted-pyramid-star-pattern.html)

[\*](https://codeforwin.org/2015/07/c-program-to-print-half-diamond-star-pattern.html)

[\*\*](https://codeforwin.org/2015/07/c-program-to-print-half-diamond-star-pattern.html)

[\*\*\*](https://codeforwin.org/2015/07/c-program-to-print-half-diamond-star-pattern.html)

[\*\*\*\*](https://codeforwin.org/2015/07/c-program-to-print-half-diamond-star-pattern.html)

[\*\*\*\*\*](https://codeforwin.org/2015/07/c-program-to-print-half-diamond-star-pattern.html)

[\*\*\*\*](https://codeforwin.org/2015/07/c-program-to-print-half-diamond-star-pattern.html)

[\*\*\*](https://codeforwin.org/2015/07/c-program-to-print-half-diamond-star-pattern.html)

[\*\*](https://codeforwin.org/2015/07/c-program-to-print-half-diamond-star-pattern.html)

[\*](https://codeforwin.org/2015/07/c-program-to-print-half-diamond-star-pattern.html)

[Half Diamond Star Pattern](https://codeforwin.org/2015/07/c-program-to-print-half-diamond-star-pattern.html)

[\*](https://codeforwin.org/2015/07/c-program-to-print-mirrored-half-diamond-star-pattern.html)

[\*\*](https://codeforwin.org/2015/07/c-program-to-print-mirrored-half-diamond-star-pattern.html)

[\*\*\*](https://codeforwin.org/2015/07/c-program-to-print-mirrored-half-diamond-star-pattern.html)

[\*\*\*\*](https://codeforwin.org/2015/07/c-program-to-print-mirrored-half-diamond-star-pattern.html)

[\*\*\*\*\*](https://codeforwin.org/2015/07/c-program-to-print-mirrored-half-diamond-star-pattern.html)

[\*\*\*\*](https://codeforwin.org/2015/07/c-program-to-print-mirrored-half-diamond-star-pattern.html)

[\*\*\*](https://codeforwin.org/2015/07/c-program-to-print-mirrored-half-diamond-star-pattern.html)

[\*\*](https://codeforwin.org/2015/07/c-program-to-print-mirrored-half-diamond-star-pattern.html)

[\*](https://codeforwin.org/2015/07/c-program-to-print-mirrored-half-diamond-star-pattern.html)

[Mirrored Half Diamond Star Pattern](https://codeforwin.org/2015/07/c-program-to-print-mirrored-half-diamond-star-pattern.html)

1. [Number pattern programs - Write a C program to print the given number patterns](https://codeforwin.org/2016/06/number-pattern-programs-in-c.html)

## Square number patterns

[11111](https://codeforwin.org/2016/06/c-program-to-print-0-1-square-number-pattern.html)

[11111](https://codeforwin.org/2016/06/c-program-to-print-0-1-square-number-pattern.html)

[11111](https://codeforwin.org/2016/06/c-program-to-print-0-1-square-number-pattern.html)

[11111](https://codeforwin.org/2016/06/c-program-to-print-0-1-square-number-pattern.html)

[11111](https://codeforwin.org/2016/06/c-program-to-print-0-1-square-number-pattern.html)

## [Number pattern 1](https://codeforwin.org/2016/06/c-program-to-print-0-1-square-number-pattern.html)

[11111](https://codeforwin.org/2016/06/c-program-to-print-number-pattern-with-0-1-at-alternate-rows.html)

[00000](https://codeforwin.org/2016/06/c-program-to-print-number-pattern-with-0-1-at-alternate-rows.html)

[11111](https://codeforwin.org/2016/06/c-program-to-print-number-pattern-with-0-1-at-alternate-rows.html)

[00000](https://codeforwin.org/2016/06/c-program-to-print-number-pattern-with-0-1-at-alternate-rows.html)

[11111](https://codeforwin.org/2016/06/c-program-to-print-number-pattern-with-0-1-at-alternate-rows.html)

## [Number pattern 2](https://codeforwin.org/2016/06/c-program-to-print-number-pattern-with-0-1-at-alternate-rows.html)

[01010](https://codeforwin.org/2016/06/c-program-to-print-number-pattern-with-1-0-at-alternate-columns.html)

[01010](https://codeforwin.org/2016/06/c-program-to-print-number-pattern-with-1-0-at-alternate-columns.html)

[01010](https://codeforwin.org/2016/06/c-program-to-print-number-pattern-with-1-0-at-alternate-columns.html)

[01010](https://codeforwin.org/2016/06/c-program-to-print-number-pattern-with-1-0-at-alternate-columns.html)

[01010](https://codeforwin.org/2016/06/c-program-to-print-number-pattern-with-1-0-at-alternate-columns.html)

## [Number pattern 3](https://codeforwin.org/2016/06/c-program-to-print-number-pattern-with-1-0-at-alternate-columns.html)

[11111](https://codeforwin.org/2016/06/c-program-to-print-box-number-pattern-with-1-0.html)

[10001](https://codeforwin.org/2016/06/c-program-to-print-box-number-pattern-with-1-0.html)

[10001](https://codeforwin.org/2016/06/c-program-to-print-box-number-pattern-with-1-0.html)

[10001](https://codeforwin.org/2016/06/c-program-to-print-box-number-pattern-with-1-0.html)

[11111](https://codeforwin.org/2016/06/c-program-to-print-box-number-pattern-with-1-0.html)

[**Number pattern 4**](https://codeforwin.org/2016/06/c-program-to-print-box-number-pattern-with-1-0.html)

[11111](https://codeforwin.org/2016/06/c-program-to-print-box-number-pattern-of-1-with-0-center.html)

[11111](https://codeforwin.org/2016/06/c-program-to-print-box-number-pattern-of-1-with-0-center.html)

[11011](https://codeforwin.org/2016/06/c-program-to-print-box-number-pattern-of-1-with-0-center.html)

[11111](https://codeforwin.org/2016/06/c-program-to-print-box-number-pattern-of-1-with-0-center.html)

[11111](https://codeforwin.org/2016/06/c-program-to-print-box-number-pattern-of-1-with-0-center.html)

## [Number pattern](https://codeforwin.org/2016/06/c-program-to-print-box-number-pattern-of-1-with-0-center.html) [5](https://codeforwin.org/2016/06/c-program-to-print-chessboard-number-pattern-of-1-0.html)

[10101](https://codeforwin.org/2016/06/c-program-to-print-chessboard-number-pattern-of-1-0.html)

[01010](https://codeforwin.org/2016/06/c-program-to-print-chessboard-number-pattern-of-1-0.html)

[10101](https://codeforwin.org/2016/06/c-program-to-print-chessboard-number-pattern-of-1-0.html)

[01010](https://codeforwin.org/2016/06/c-program-to-print-chessboard-number-pattern-of-1-0.html)

[10101](https://codeforwin.org/2016/06/c-program-to-print-chessboard-number-pattern-of-1-0.html)

# If…Else Exercises

1. [Write a C program to find maximum between two numbers.](https://codeforwin.org/2015/05/c-program-to-find-maximum-between-two-numbers.html)
2. [Write a C program to find maximum between three numbers.](https://codeforwin.org/2015/05/c-program-to-find-maximum.html)
3. [Write a C program to check whether a number is negative, positive or zero.](https://codeforwin.org/2015/05/c-program-to-check-negative-positive-zero.html)
4. [Write a C program to check whether a number is divisible by 5 and 11 or not.](https://codeforwin.org/2015/05/c-program-to-check-whether-number-is-divisible-by-5-and-11.html)
5. [Write a C program to check whether a number is even or odd.](https://codeforwin.org/2015/05/c-program-to-check-even-odd.html)
6. [Write a C program to check whether a year is leap year or not.](https://codeforwin.org/2015/05/c-program-to-check-leap-year.html)
7. [Write a C program to check whether a character is alphabet or not.](https://codeforwin.org/2015/05/c-program-to-check-alphabet.html)
8. [Write a C program to input any alphabet and check whether it is vowel or consonant.](https://codeforwin.org/2015/05/c-program-to-check-vowel-or-consonant.html)
9. [Write a C program to input any character and check whether it is alphabet, digit or special character.](https://codeforwin.org/2015/05/c-program-to-check-alphabet-digit-special-character.html)
10. [Write a C program to check whether a character is uppercase or lowercase alphabet.](https://codeforwin.org/2015/09/c-program-to-check-whether-character-is-uppercase-or-lowercase.html)
11. [Write a C program to input week number and print week day.](https://codeforwin.org/2015/05/c-program-to-print-week-name.html)
12. [Write a C program to input month number and print number of days in that month.](https://codeforwin.org/2015/05/c-program-to-print-number-of-days-in-month.html)
13. [Write a C program to count total number of notes in given amount](https://codeforwin.org/2015/07/c-program-to-count-total-number-of-notes.html).
14. [Write a C program to input month number and print number of days in that month.](https://codeforwin.org/2015/05/c-program-to-print-number-of-days-in-month.html)
15. [Write a C program to count total number of notes in given amount](https://codeforwin.org/2015/07/c-program-to-count-total-number-of-notes.html).
16. [Write a C program to input angles of a triangle and check whether triangle is valid or not.](https://codeforwin.org/2015/05/c-program-to-check-whether-triangle-is-valid-or-not.html)
17. [Write a C program to input all sides of a triangle and check whether triangle is valid or not.](https://codeforwin.org/2015/05/c-program-to-check-whether-triangle-side-is-valid-or-not.html)
18. [Write a C program to check whether the triangle is equilateral, isosceles or scalene triangle.](https://codeforwin.org/2015/05/c-program-to-check-whether-triangle-is-equilateral-isosceles-scalene.html)
19. [Write a C program to find all roots of a quadratic equation.](https://codeforwin.org/2016/04/c-program-to-find-all-roots-of-quadratic-equation.html)
20. [Write a C program to calculate profit or loss.](https://codeforwin.org/2015/05/c-program-to-check-profit-or-loss.html)
21. [Write a C program to input marks of five subjects Physics, Chemistry, Biology, Mathematics and Computer. Calculate percentage and grade according to following:](https://codeforwin.org/2015/05/c-program-to-enter-student-marks-and-calculate-percentage-and-grade.html)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Percentage | >= | 90% : | Grade | A |
| Percentage | >= | 80% : | Grade | B |
| Percentage | >= | 70% : | Grade | C |
| Percentage | >= | 60% : | Grade | D |
| Percentage | >= | 40% : | Grade | E |

Percentage < 40% : Grade F

1. [Write a C program to input basic salary of an employee and calculate its Gross salary according to following:](https://codeforwin.org/2015/05/c-program-to-calculate-gross-salary-of-employee.html)

Basic Salary <= 10000 : HRA = 20%, DA = 80% Basic Salary <= 20000 : HRA = 25%, DA = 90% Basic Salary > 20000 : HRA = 30%, DA = 95%

1. [Write a C program to input electricity unit charges and calculate total electricity bill according to](https://codeforwin.org/2015/05/c-program-to-calculate-electricity-bill.html)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| [the](https://codeforwin.org/2015/05/c-program-to-calculate-electricity-bill.html)  For | first | 50 | [given](https://codeforwin.org/2015/05/c-program-to-calculate-electricity-bill.html) | units | Rs. | [condition:](https://codeforwin.org/2015/05/c-program-to-calculate-electricity-bill.html)  0.50/unit |
| For | next | 100 |  | units | Rs. | 0.75/unit |
| For | next | 100 |  | units | Rs. | 1.20/unit |
| For | unit | above |  | 250 | Rs. | 1.50/unit |

An additional surcharge of 20% is added to the bill

# C Functions Exercises

1. [Write a C program to find cube of any number using function.](https://codeforwin.org/2016/02/c-program-to-find-cube-of-number-using-function.html)
2. [Write a C program to find diameter, circumference and area of circle using functions](https://codeforwin.org/2016/02/c-program-to-find-diameter-circumference-and-area-of-circle-using-function.html).
3. [Write a C program to find maximum and minimum between two numbers using functions.](https://codeforwin.org/2016/02/c-program-to-find-maximum-and-minimum-using-functions.html)
4. [Write a C program to check whether a number is even or odd using functions](https://codeforwin.org/2016/02/c-program-to-check-even-or-odd-using-functions.html).
5. [Write a C program to check whether a number is Prime, Armstrong or perfect number using functions.](https://codeforwin.org/2016/02/c-program-to-check-prime-armstrong-perfect-number-using-functions.html)
6. [Write a C program to find all prime numbers between given interval using functions.](https://codeforwin.org/2016/02/c-program-to-find-prime-numbers-between-two-intervals-using-function.html)
7. [Write a C program to print all strong numbers between given interval using functions](https://codeforwin.org/2016/03/c-program-to-find-all-strong-numbers-between-two-interval-using-function.html).
8. [Write a C program to print all Armstrong numbers between given interval using functions.](https://codeforwin.org/2016/03/c-program-to-print-armstrong-numbers-between-1-to-n-using-function.html)
9. [Write a C program to print all perfect numbers between given interval using functions](https://codeforwin.org/2016/03/c-program-to-print-all-perfect-numbers-between-1-to-n-using-function.html).
10. [Write a C program to find power of any number using recursion](https://codeforwin.org/2016/02/c-program-to-find-power-of-any-number-using-recursion.html).
11. [Write a C program to print all natural numbers between 1 to n using recursion.](https://codeforwin.org/2016/02/c-program-to-print-all-natural-numbers-in-given-range-using-recursion.html)
12. [Write a C program to print all even or odd numbers in given range using recursion.](https://codeforwin.org/2016/03/c-program-to-print-even-odd-numbers-using-recursion.html)
13. [Write a C program to find sum of all natural numbers between 1 to n using recursion](https://codeforwin.org/2016/02/c-program-to-find-sum-of-natural-numbers-using-recursion.html).
14. [Write a C program to find sum of all even or odd numbers in given range using recursion.](https://codeforwin.org/2016/03/c-program-to-find-sum-of-even-odd-number-using-recursion.html)
15. [Write a C program to find reverse of any number using recursion.](https://codeforwin.org/2016/03/c-program-to-find-reverse-of-number-using-recursion.html)
16. [Write a C program to check whether a number is palindrome or not using recursion](https://codeforwin.org/2016/03/c-program-to-check-palindrome-number-using-recursion.html).
17. [Write a C program to find sum of digits of a given number using recursion.](https://codeforwin.org/2016/03/c-program-to-calculate-sum-of-digits-using-recursion.html)
18. [Write a C program to find factorial of any number using recursion.](https://codeforwin.org/2016/02/c-program-to-find-factorial-of-number-using-recursive-function.html)
19. [Write a C program to generate nth Fibonacci term using recursion.](https://codeforwin.org/2016/02/c-program-to-generate-nth-fibonacci-series-using-recursion.html)
20. [Write a C program to find GCD (HCF) of two numbers using recursion](https://codeforwin.org/2016/03/c-program-to-find-gcd-of-two-numbers-using-recursion.html).
21. [Write a C program to find LCM of two numbers using recursion.](https://codeforwin.org/2016/03/c-program-to-find-lcm-of-two-numbers-using-recursion.html)

1.write a c program to print all natural numbers from 1 to n

#include <stdio.h>

int main()

{

int i,n;

printf("enter value of n:");

scanf("%d",&n);

for(i=1;i<=n;i++)

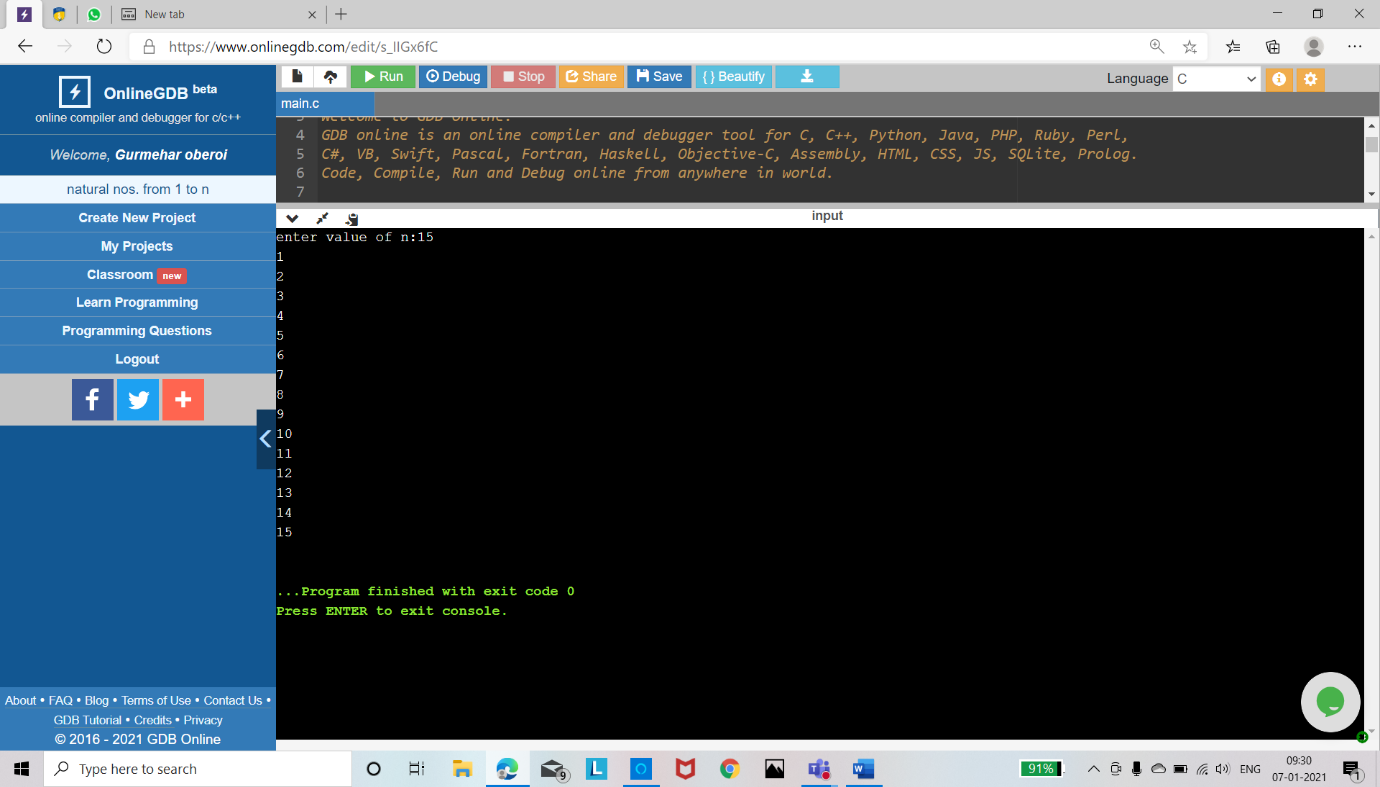
{

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

}

return 0;

}



2. Write a c program to print all natural numbers in reverse from n to 1

#include <stdio.h>

int main()

{

int i,n;

printf("enter value of n:");

scanf("%d",&n);

for(i=n;i>=1;i--)

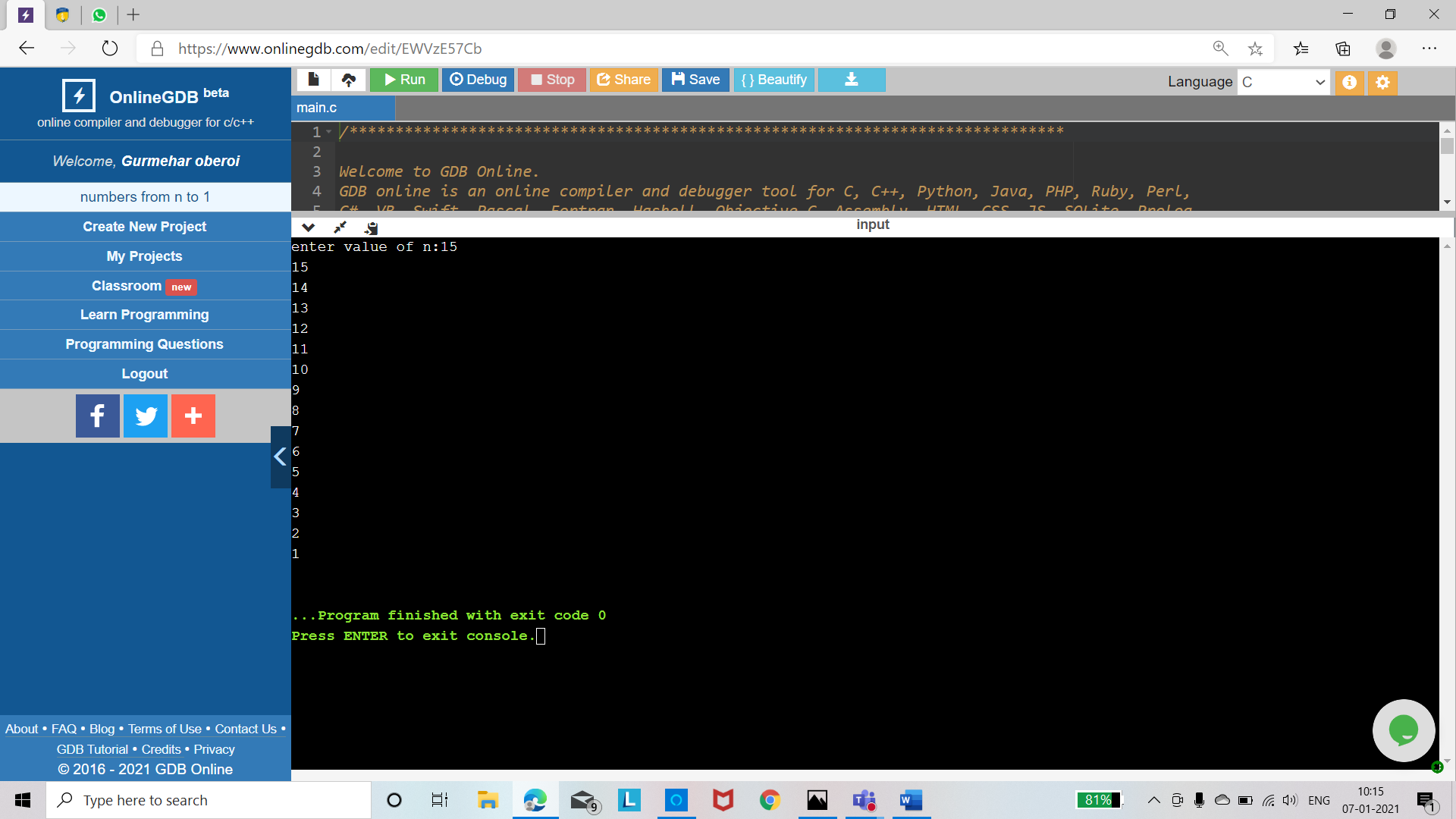
{

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

}

return 0;

}



3.write a program to print all alphabets from a to z

#include <stdio.h>

int main()

{

char ch;

printf("Alphabets from a - z are: \n");

for(ch='a'; ch<='z'; ch++)

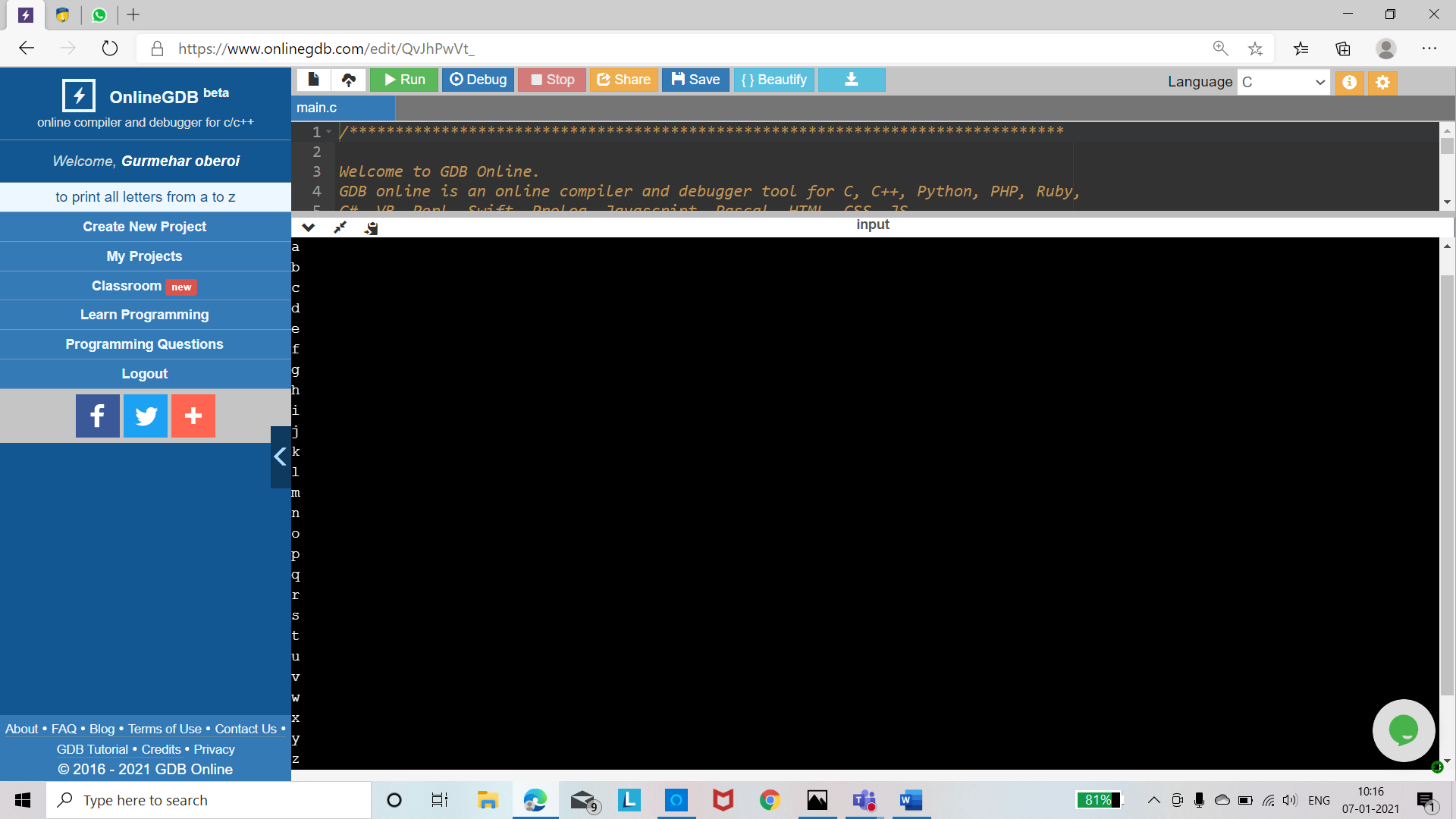
{

printf("%c\n", ch);

}

return 0;

}



4.write a program to print all even numbers between 1 to 100

int main()

{

int i;

for(i=2;i<=100;i=i+2)

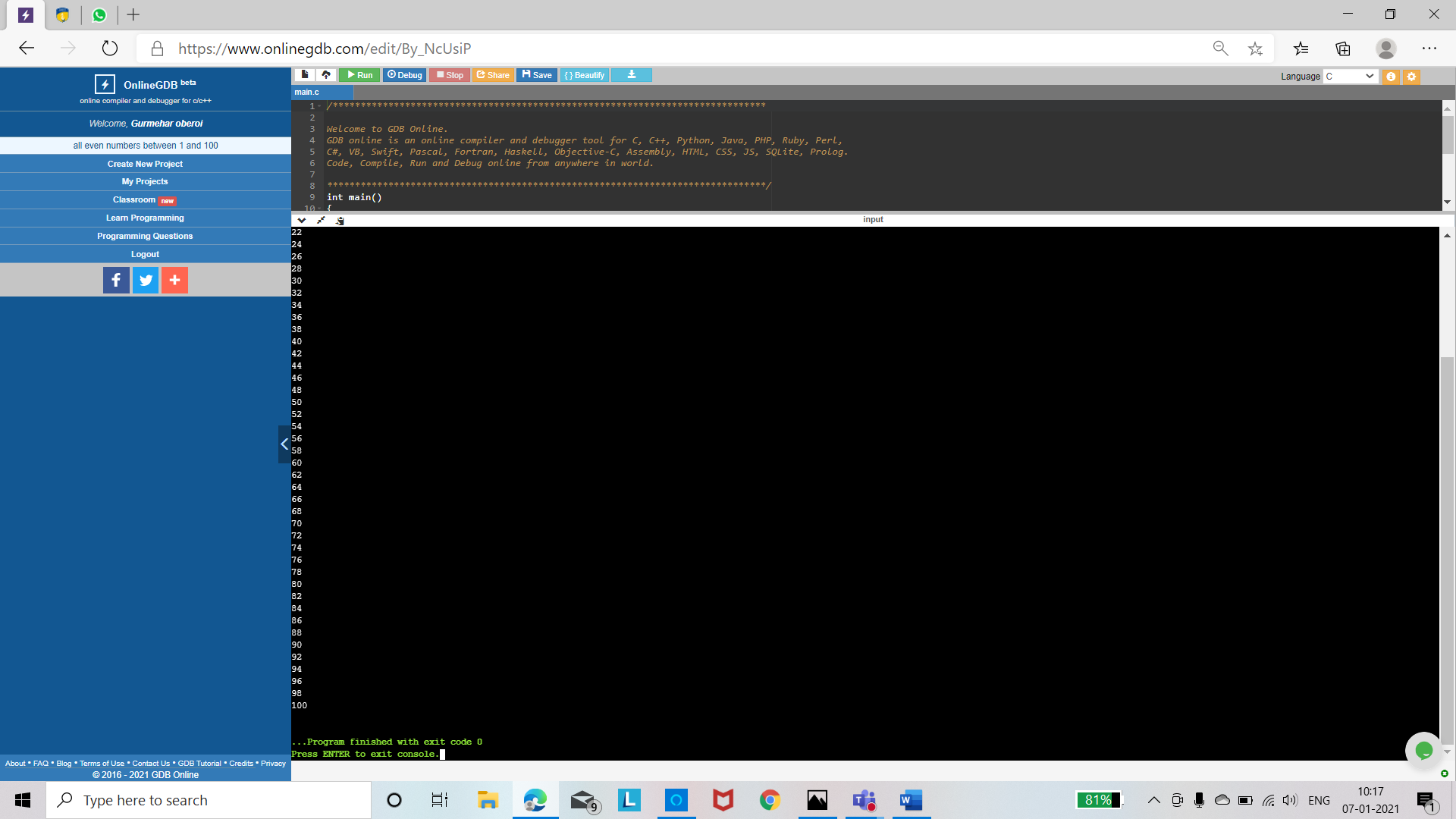
{

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

}

return 0;

}



5.write a program to print all odd numbers from 1 to 100

#include <stdio.h>

int main()

{

int i;

for(i=1;i<=100;i=i+2)

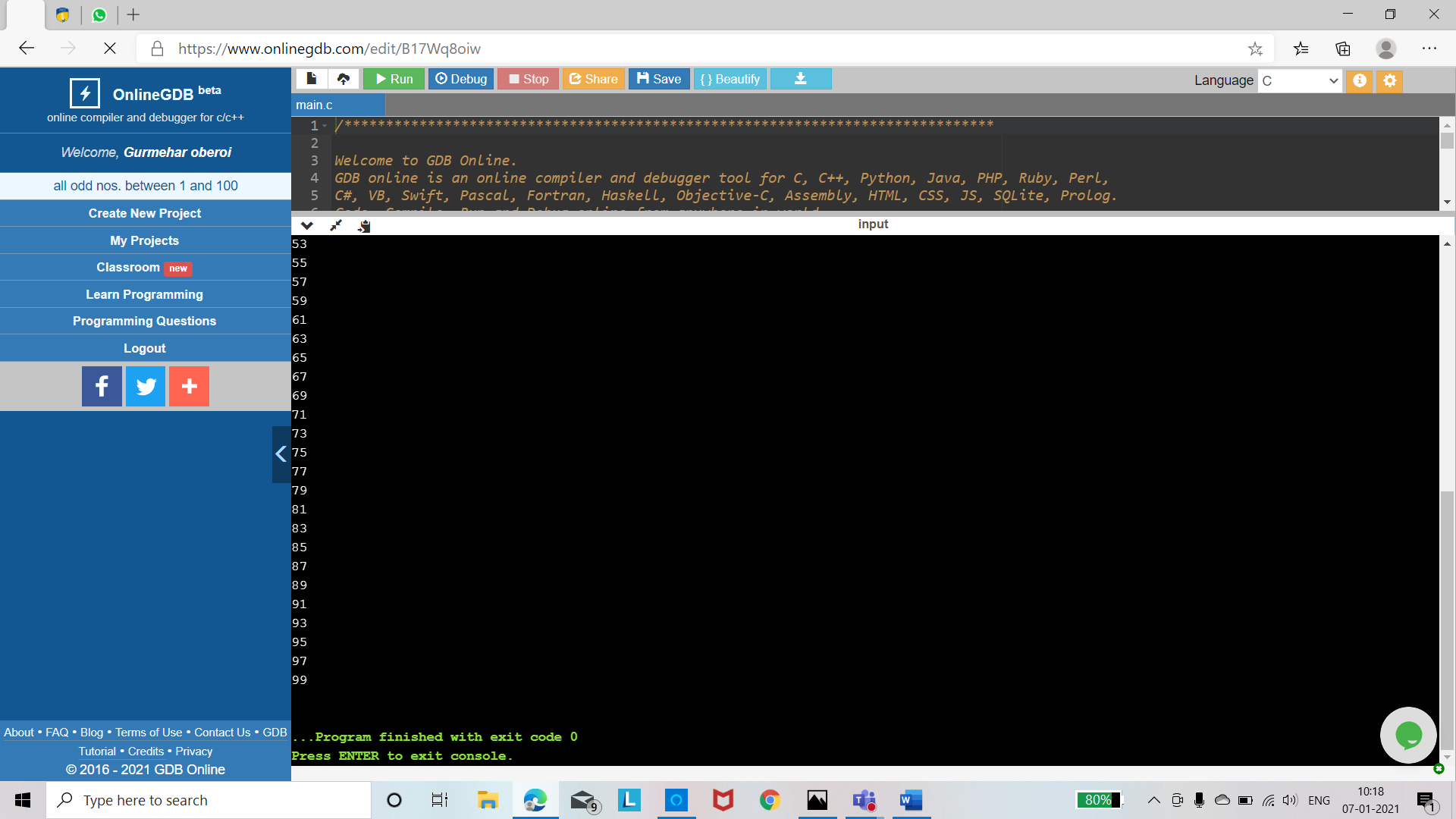
{

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

}

return 0;

}



6.write a program to find sum of all natural numbers between 1 to n

#include<stdio.h>

int main()

{

int Number, i, Sum = 0;

printf("\nPlease Enter any Integer Value\n");

scanf("%d", &Number);

for(i = 1; i <= Number; i++)

{

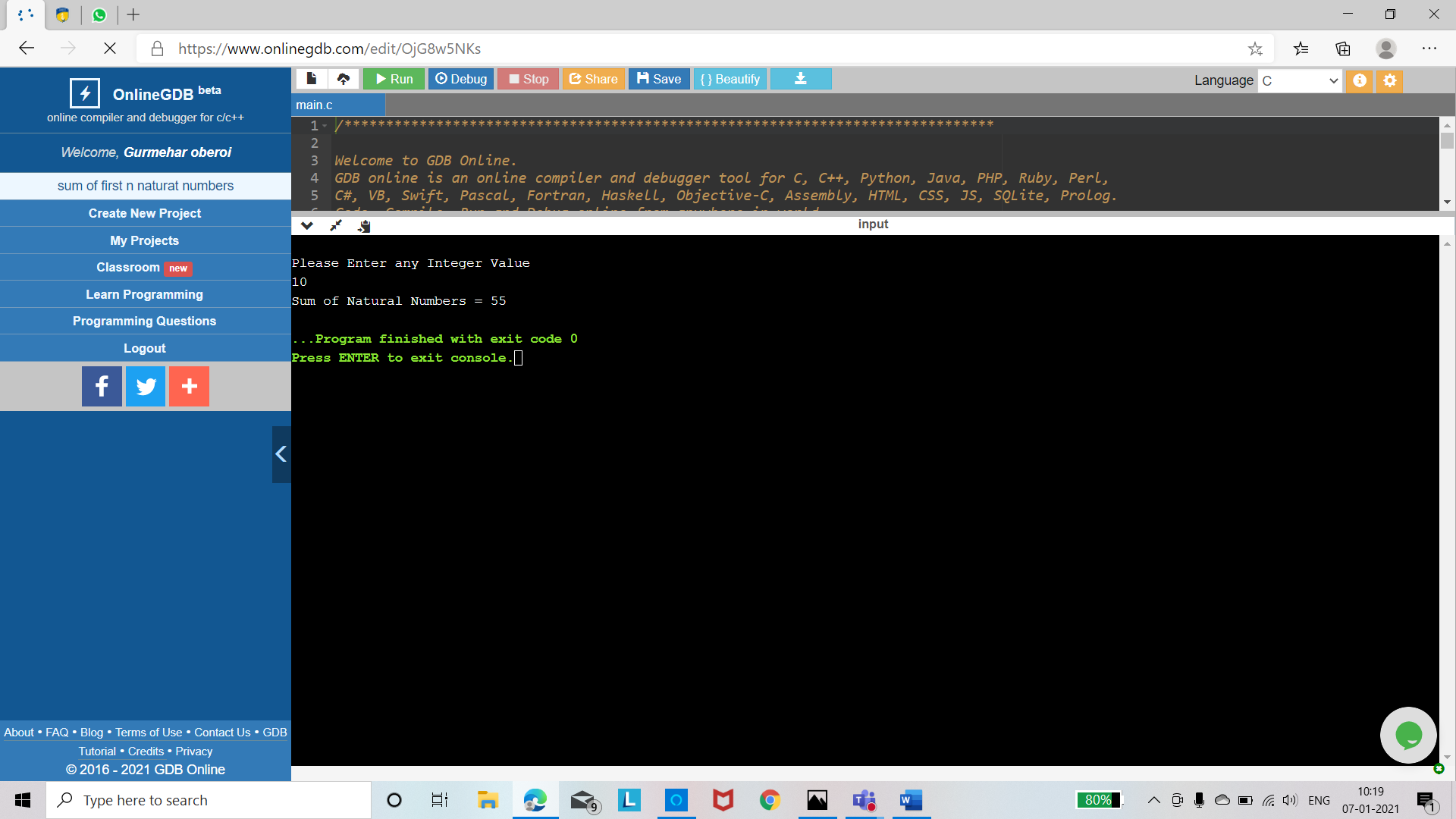
Sum = Sum + i;

}

printf("Sum of Natural Numbers = %d", Sum);

return 0;

}



7.write a program to find sum of all even numbers between 1 to n

#include<stdio.h>

int main()

{

int i, number, a;

printf("\n Please Enter the Maximum Even Value : ");

scanf("%d", &number);

for(i = 1; i <= number; i++)

{

if ( i%2 == 0 )

{

a = a + i;

}

else

{

printf("error");

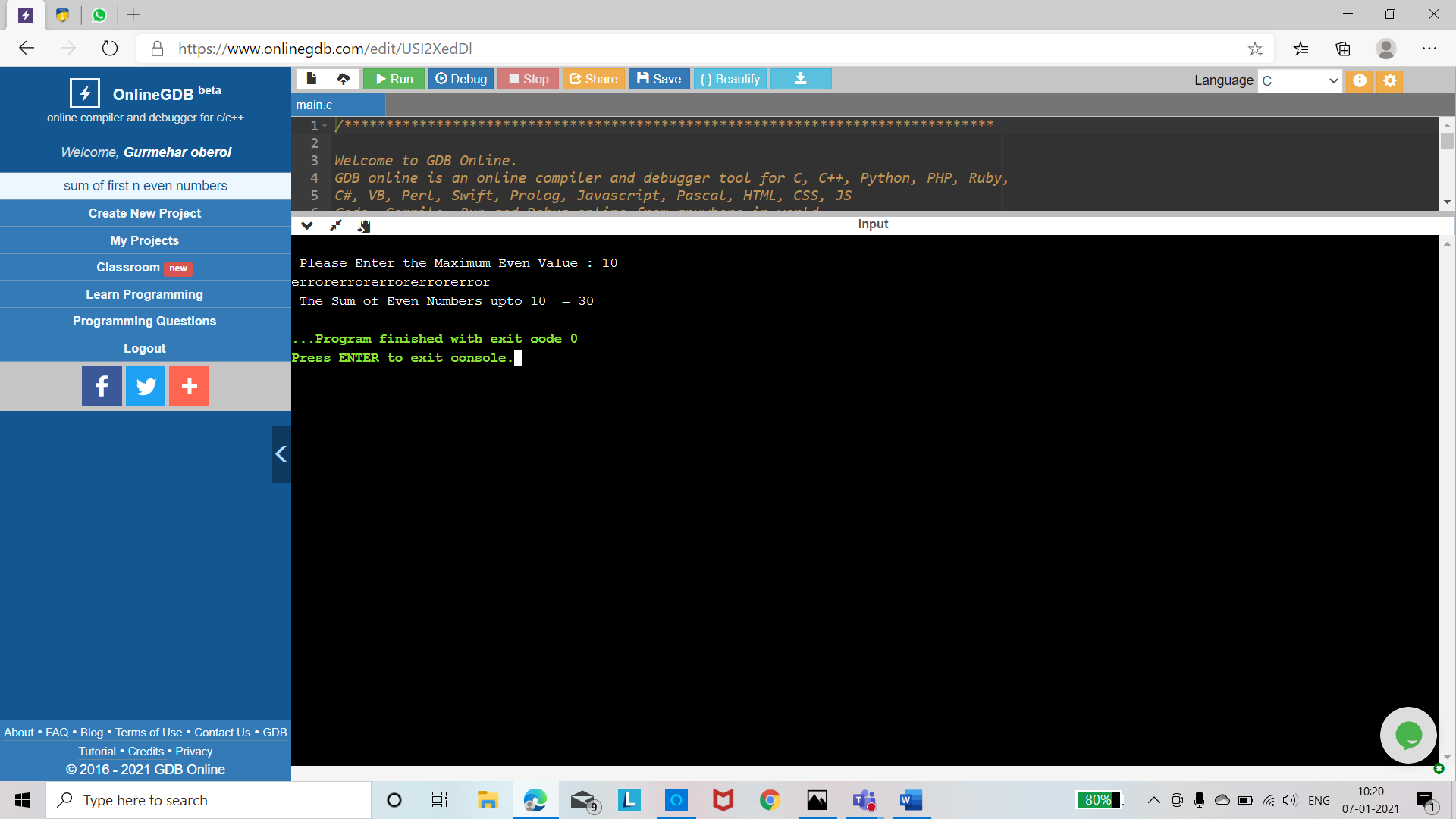
}

}

printf("\n The Sum of Even Numbers upto %d = %d", number, a);

return 0;

}



8. write a program to find sum of all odd numbers between 1 to n

#include<stdio.h>

int main()

{

int i, number, a;

printf("\n Please Enter the Maximum Value : ");

scanf("%d", &number);

for(i = 1; i <= number; i++)

{

if ( i%2 == 1 )

{

a = a + i;

}

else

{

printf("error");

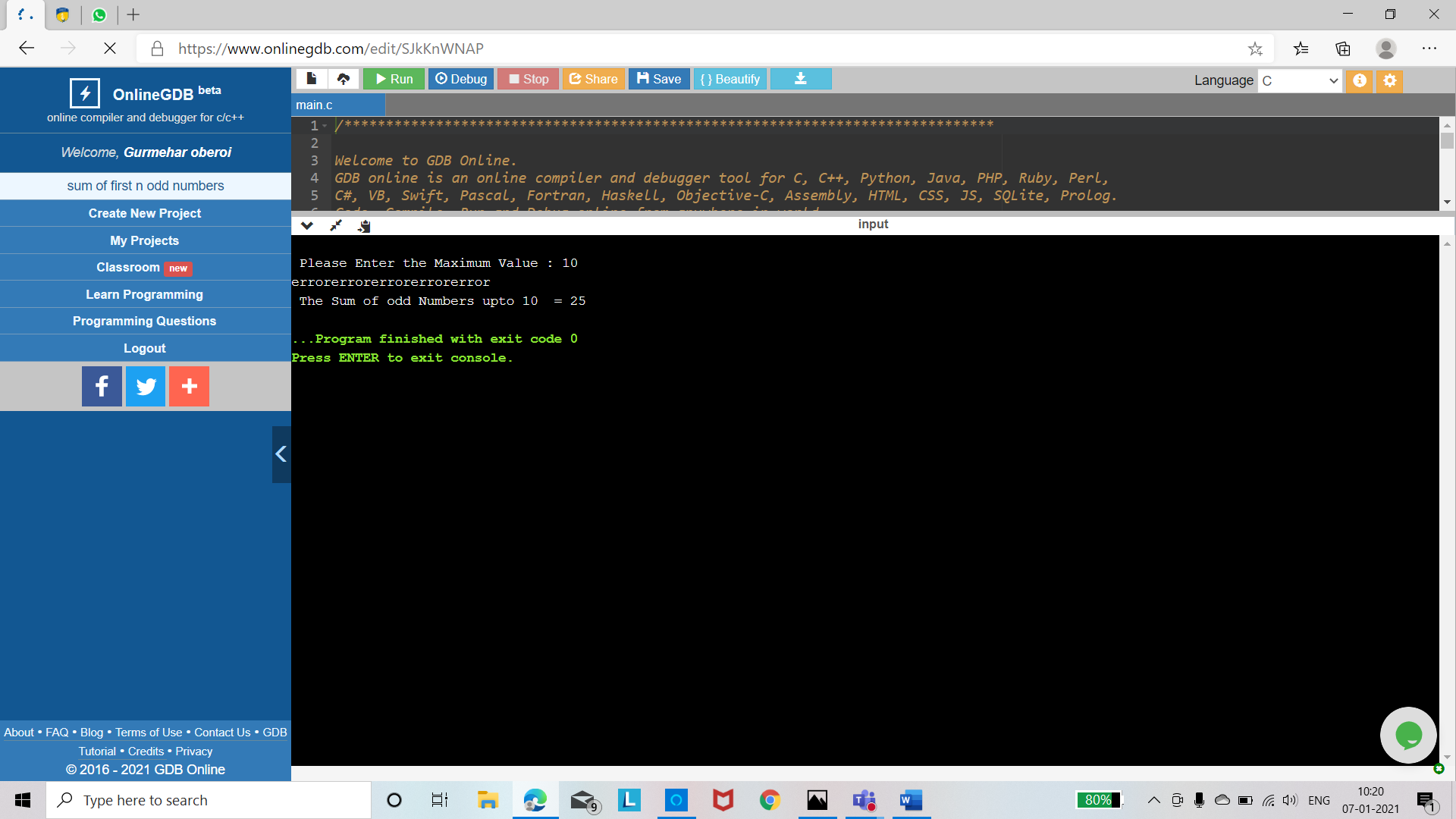
}

}

printf("\n The Sum of odd Numbers upto %d = %d", number, a);

return 0;

}



9.write a program to print a multiplication table of a given number n

#include <stdio.h>

int main() {

int number, i;

printf("Please enter any number to find multiplication table:");

scanf("%d", &number);

printf("Multiplication table for the given number %d: ", number);

printf("\n");

for(i=1;i<=10;i++){

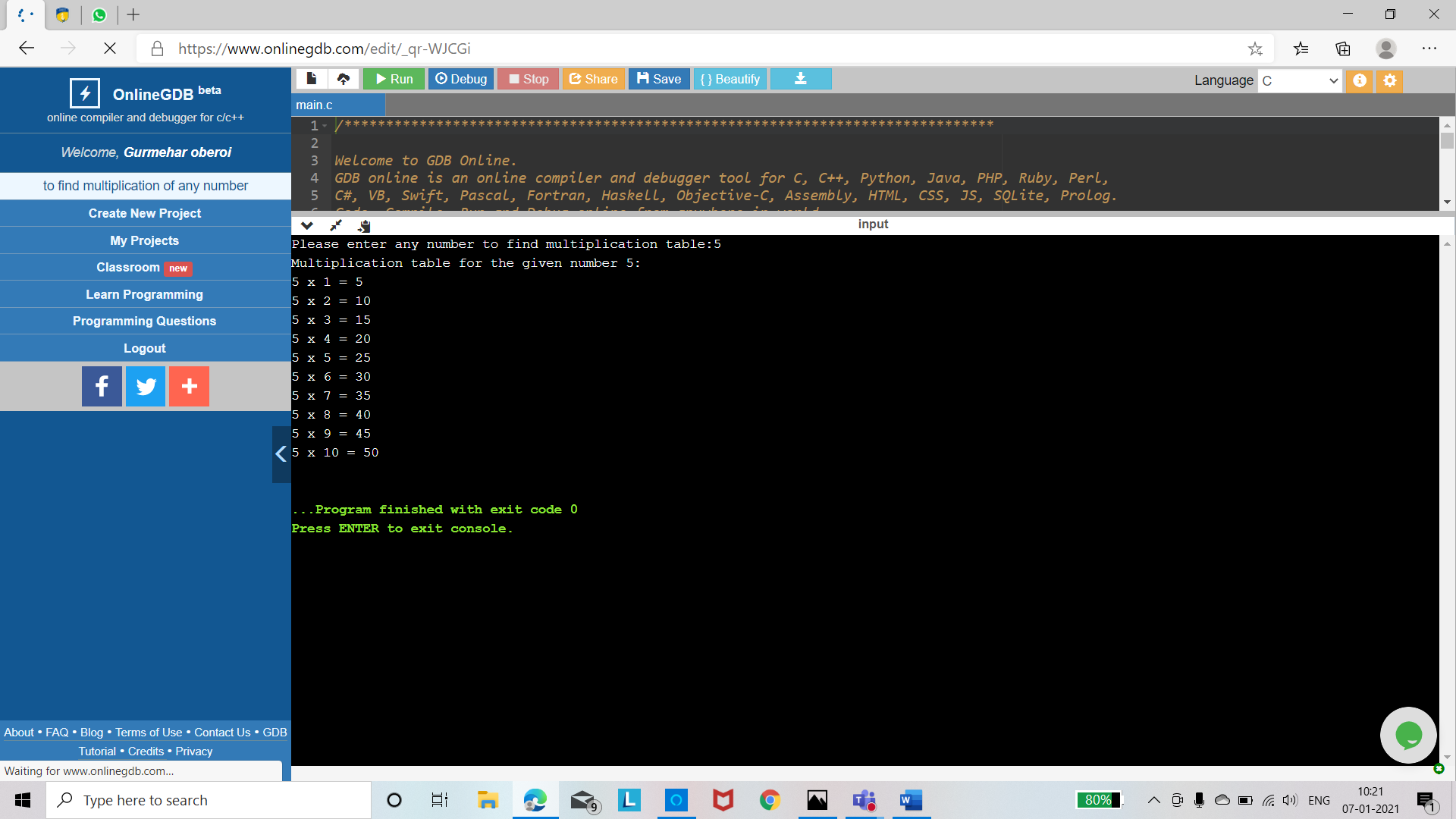
printf("%d x %d = %d", number, i, number \* i);

printf("\n");

}

return 0;

}



10. Write a C program to count number of digits in a number.

#include <stdio.h>

int main()

{

long long num;

int count=0;

printf("enter the number: ");

scanf("%lld",&num);

do

{

count++;

num/=10;

}while(num!=0);

printf("total number of digits in the added number:%d",count);

return 0;

}

enter the number: 76453892

total number of digits in the added number:8

**...Program finished with exit code 0**

**Press ENTER to exit console.**

11. Write a C program to find first and last digit of a number.

#include <stdio.h>

#include <math.h>

int main()

{

int n, firstDigit, lastDigit, digits;

printf("Enter any number: ");

scanf("%d", &n);

lastDigit = n % 10;

digits = (int)log10(n);

firstDigit = (int)(n / pow(10, digits));

printf("First digit = %d\n", firstDigit);

printf("Last digit = %d\n", lastDigit);

return 0;

}

Enter any number: 24367

First digit = 2

Last digit = 7

**...Program finished with exit code 0**

**Press ENTER to exit console.**

12. Write a C program to find sum of first and last digit of a number.

#include <stdio.h>

#include <math.h>

int main()

{

int n, firstDigit, lastDigit, digits;

printf("Enter any number: ");

scanf("%d", &n);

lastDigit = n % 10;

digits = (int)log10(n);

firstDigit = (int)(n / pow(10, digits));

printf("sum of first and last digit = %d\n",firstDigit + lastDigit);

return 0;

}

Enter any number: 53627

sum of first and last digit = 12

**...Program finished with exit code 0**

**Press ENTER to exit console.**

13. Write a C program to swap first and last digits of a number.

#include <stdio.h>

#include <math.h>

int main()

{

int n, firstDigit, lastDigit, digits, temp;

printf("Enter any number: ");

scanf("%d", &n);

lastDigit = n % 10;

digits = (int)log10(n);

firstDigit = (int)(n / pow(10, digits));

temp=firstDigit;

firstDigit=lastDigit;

lastDigit=temp;

printf("swapped firstdigit = %d\n",firstDigit);

printf("swapped lastdigit = %d\n",lastDigit);

return 0;

}

Enter any number: 675834

swapped firstdigit = 4

swapped lastdigit = 6

**...Program finished with exit code 0**

**Press ENTER to exit console.**

14. Write a C program to calculate sum of digits of a number.

#include <stdio.h>

#include <math.h>

int main()

{

int n,sum;

printf("enter the number:");

scanf("%d",&n);

while(n!=0)

{

sum+=n % 10;

n/=10;

}

printf("the sum of the digits:%d",sum);

return 0;

}

enter the number:1234

the sum of the digits:10

**...Program finished with exit code 0**

**Press ENTER to exit console.**

15. Write a C program to calculate product of digits of a number.

#include <stdio.h>

#include <math.h>

int main()

{

int n,product=1;

printf("enter the number:");

scanf("%d",&n);

while(n!=0)

{

product\*=n % 10;

n/=10;

}

printf("the product of the digits:%d",product);

return 0;

}

enter the number:1234

the product of the digits:24

**...Program finished with exit code 0**

**Press ENTER to exit console.**

16. Write a C program to enter a number and print its reverse.

#include <stdio.h>

#include <math.h>

int main()

{

int num, reverse = 0;

printf("Enter any number to find reverse: ");

scanf("%d", &num);

while(num != 0)

{

reverse = (reverse \* 10) + (num % 10);

num /= 10;

}

printf("Reverse = %d", reverse);

return 0;

}

Enter any number to find reverse: 12345

Reverse = 54321

**...Program finished with exit code 0**

**Press ENTER to exit console.**

17. Write a C program to check whether a number is palindrome or not.

#include <stdio.h>

int main()

{

int n, num, rev = 0;

printf("Enter any number to check palindrome: ");

scanf("%d", &n);

num = n;

while(n != 0)

{

rev = (rev \* 10) + (n % 10);

n /= 10;

}

if(rev == num)

{

printf("%d is palindrome.", num);

}

else{

printf("%d is not palindrome.", num);

}

return 0;}

Enter any number to find reverse: 12345

Reverse = 54321

**...Program finished with exit code 0**

**Press ENTER to exit console.**

18. Write a C program to find frequency of each digit in a given integer.

#include <stdio.h>

#define BASE 10

int main()

{

long long num, n;

int i, lastDigit;

int freq[BASE];

printf("Enter any number: ");

scanf("%lld", &num);

for(i=0; i<BASE; i++)

{

freq[i] = 0;

}

n = num;

while(n != 0)

{

lastDigit = n % 10;

n /= 10;

freq[lastDigit]++;

}

printf("Frequency of each digit in %lld is: \n", num);

for(i=0; i<BASE; i++)

{

printf("Frequency of %d = %d\n", i, freq[i]);

}

return 0;

}

Enter any number: 1234

Frequency of each digit in 1234 is:

Frequency of 0 = 0

Frequency of 1 = 1

Frequency of 2 = 1

Frequency of 3 = 1

Frequency of 4 = 1

Frequency of 5 = 0

Frequency of 6 = 0

Frequency of 7 = 0

Frequency of 8 = 0

Frequency of 9 = 0

**...Program finished with exit code 0**

**Press ENTER to exit console.**

19.Write a program to enter a number and print it in words:

#include <stdio.h>

int main()

{

int n, num = 0;

printf("Enter any number to print in words: ");

scanf("%d", &n);

while(n != 0)

{

num = (num \* 10) + (n % 10);

n /= 10;

}

while(num != 0)

{

switch(num % 10)

{

case 0:

printf("Zero ");

break;

case 1:

printf("One ");

break;

case 2:

printf("Two ");

break;

case 3:

printf("Three ");

break;

case 4:

printf("Four ");

break;

case 5:

printf("Five ");

break;

case 6:

printf("Six ");

break;

case 7:

printf("Seven ");

break;

case 8:

printf("Eight ");

break;

case 9:

printf("Nine ");

break;

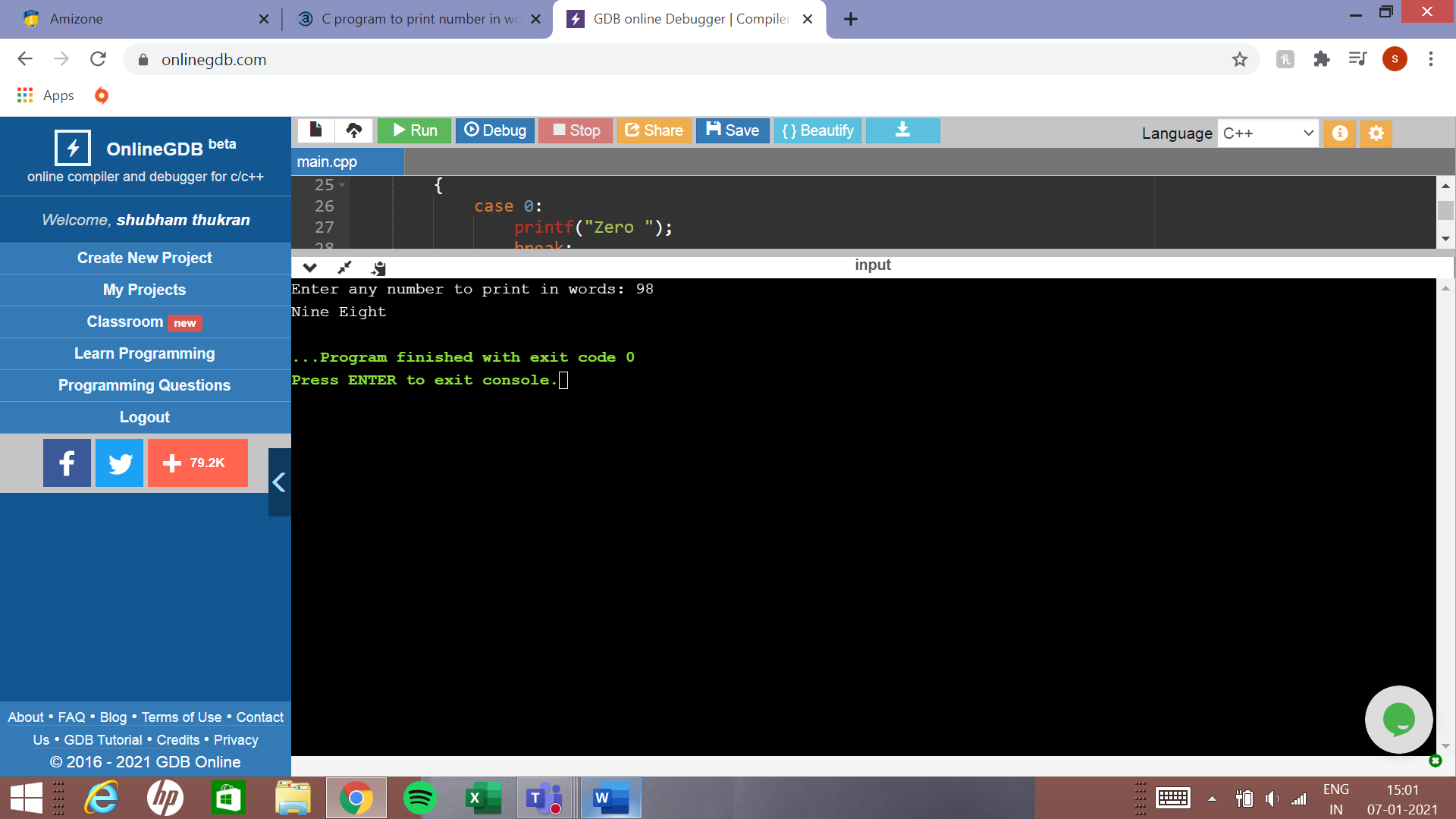
}

num = num / 10;

}

return 0;

}



20.Write a program to print all ASCII character with their values:

#include <stdio.h>

int main()

{

int i;

for(i=0; i<=255; i++)

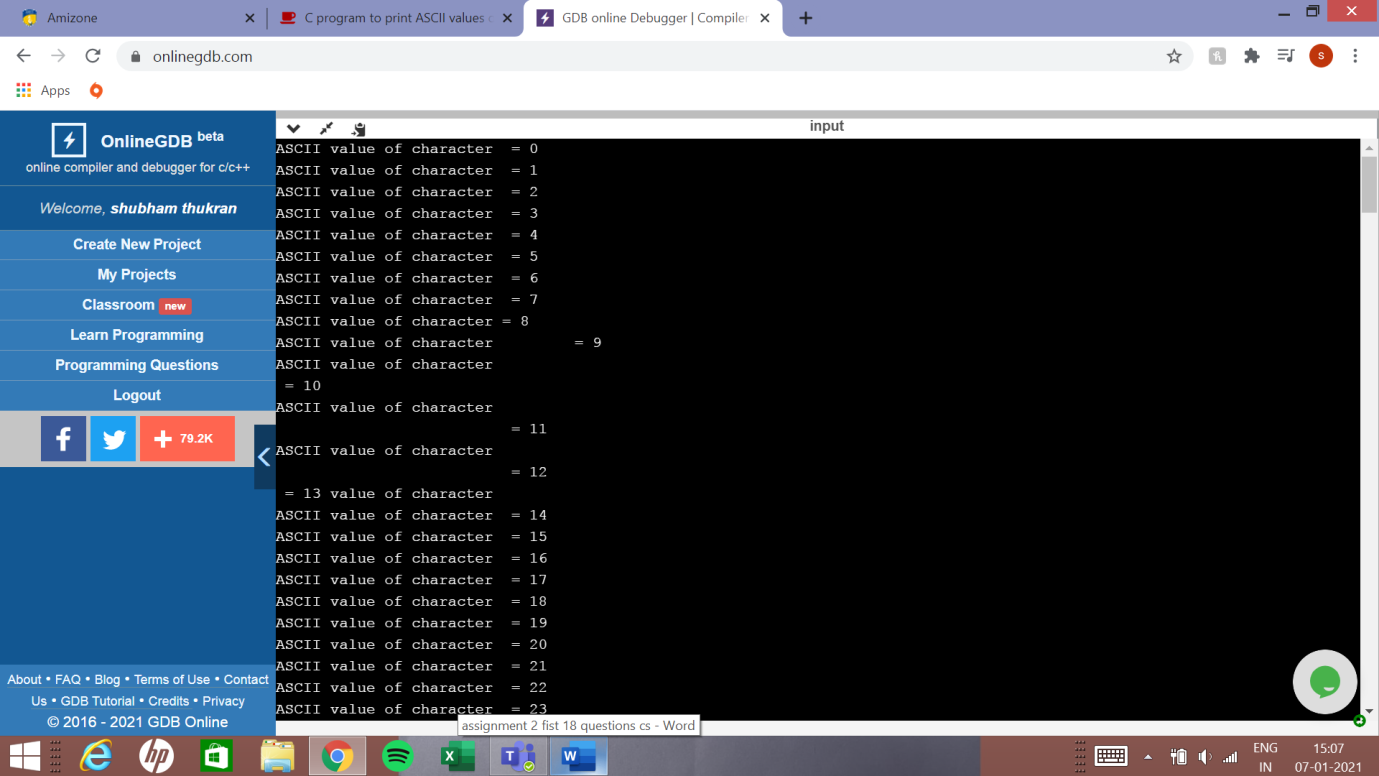
{

printf("ASCII value of character %c = %d\n", i, i);

}

return 0;

}



21.write a program to find power of a number using for loop:

#include <stdio.h>

int main()

{

int i, number, exponent;

long power = 1;

printf("\n Please Enter any number : ");

scanf(" %d", &number);

printf("\n Please Enter the exponent value: ");

scanf(" %d", &exponent);

for(i = 1; i <= exponent; i++)

{

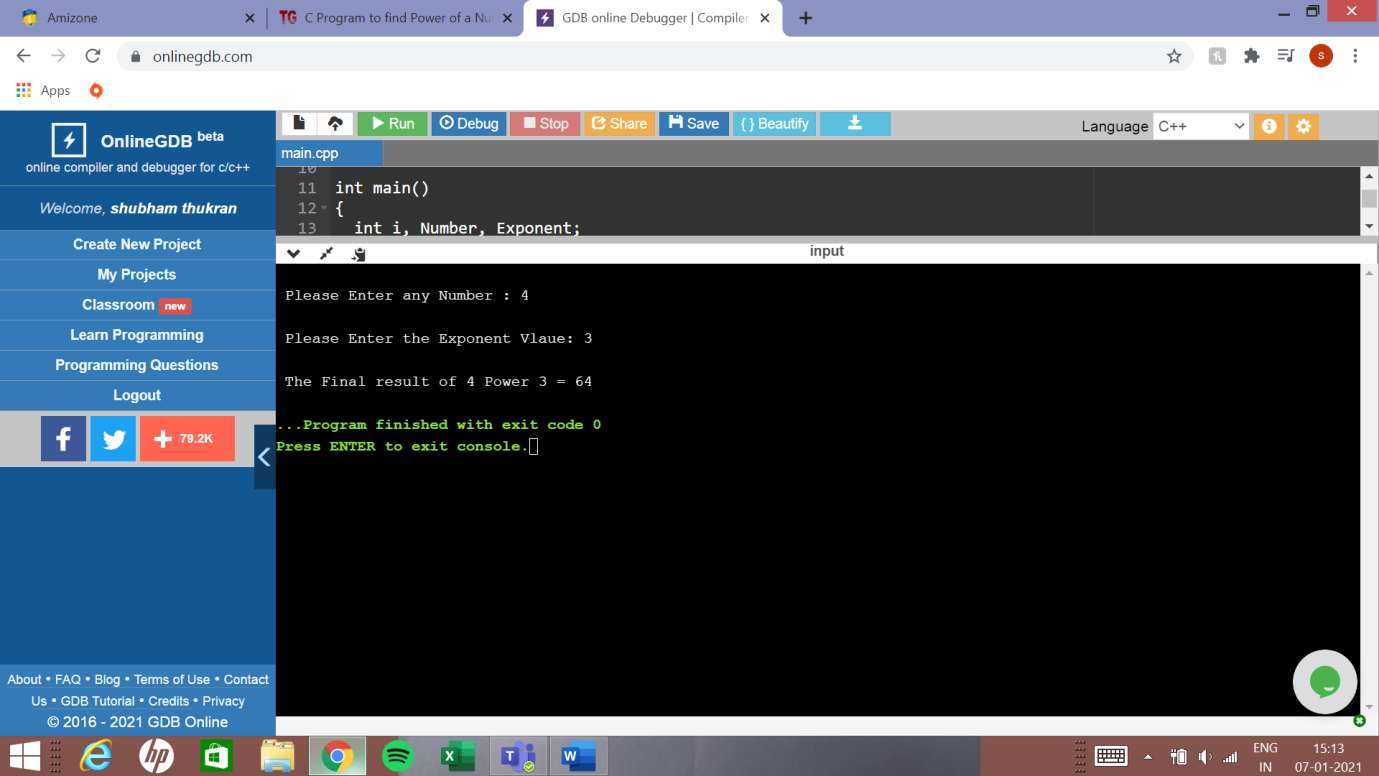
power = power \* number;

}

printf("\n The Final result of %d Power %d = %ld", number, exponent, power);

return 0;

}



22.Write a program to find all factors of a number:

#include <stdio.h>

int main() {

int num, i;

printf("Enter a positive integer: ");

scanf("%d", &num);

printf("Factors of %d are: ", num);

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

if (num % i == 0) {

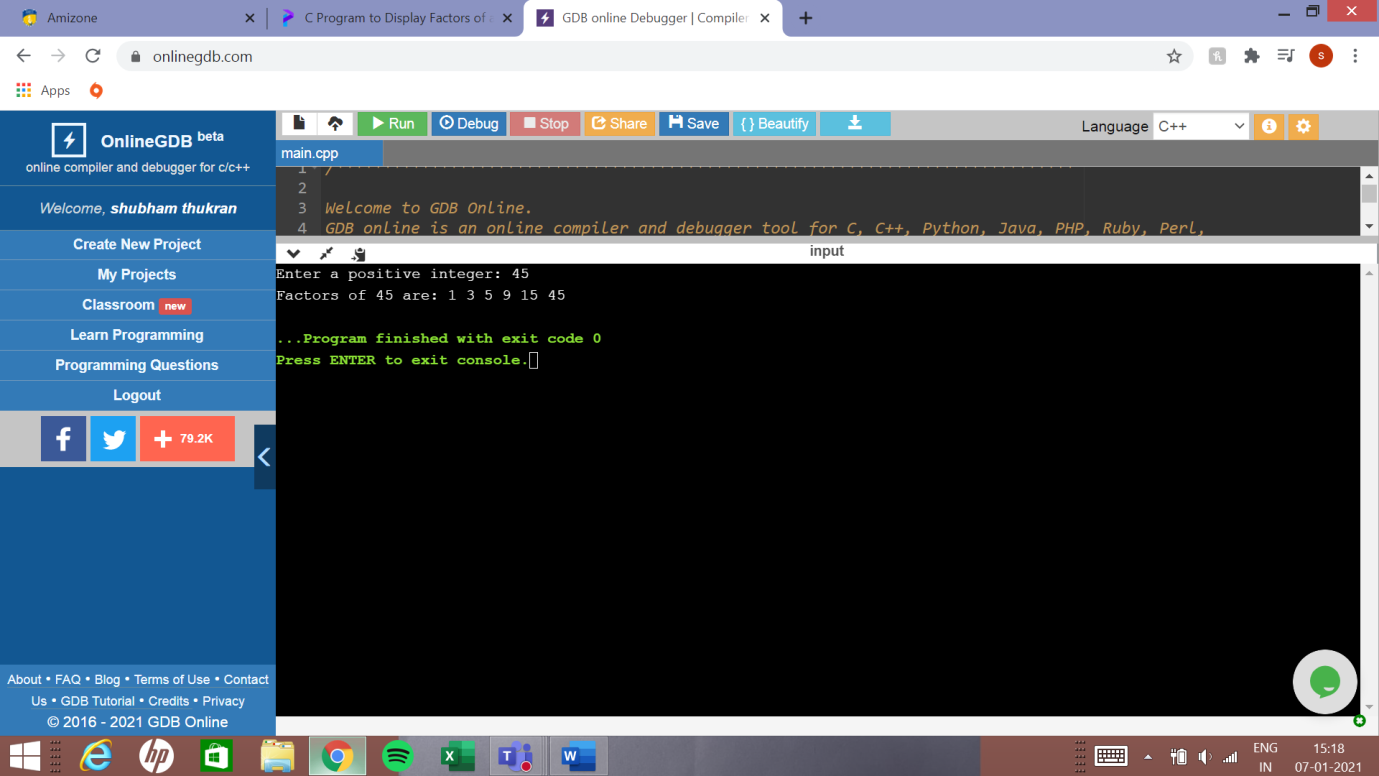
printf("%d ", i);

}

}

return 0;

}



23.Write a program to calculate factorial of a number:

#include <stdio.h>

int main() {

int n, i;

unsigned long long fact = 1;

printf("Enter an integer: ");

scanf("%d", &n);

if (n < 0)

printf("Error! Factorial of a negative number doesn't exist.");

else {

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

fact \*= i;

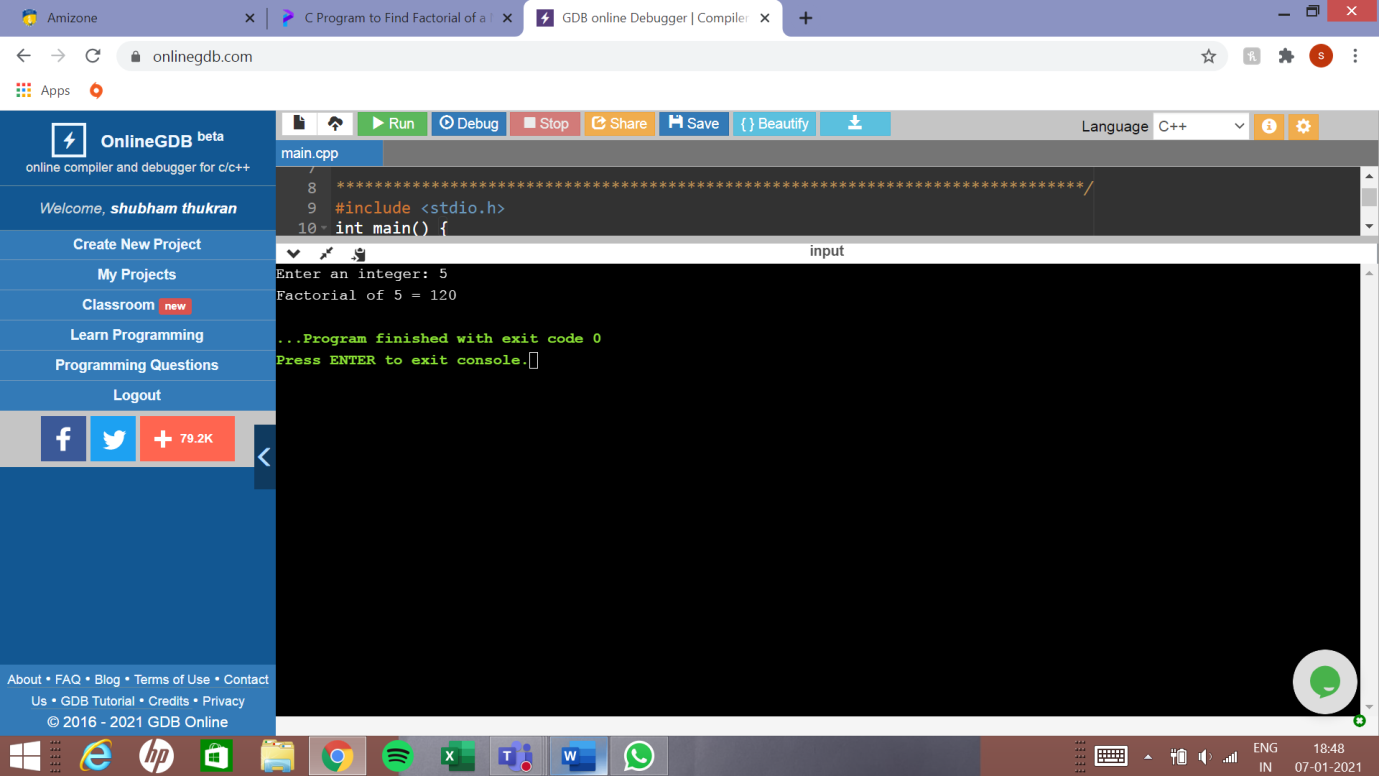
}

printf("Factorial of %d = %llu", n, fact);

}

return 0;

}



24.Write a program to find HCF(GCD) of two numbers:

#include <stdio.h>

int main()

{

int n1, n2, i, gcd;

printf("Enter two integers: ");

scanf("%d %d", &n1, &n2);

for(i=1; i <= n1 && i <= n2; ++i)

{

if(n1%i==0 && n2%i==0)

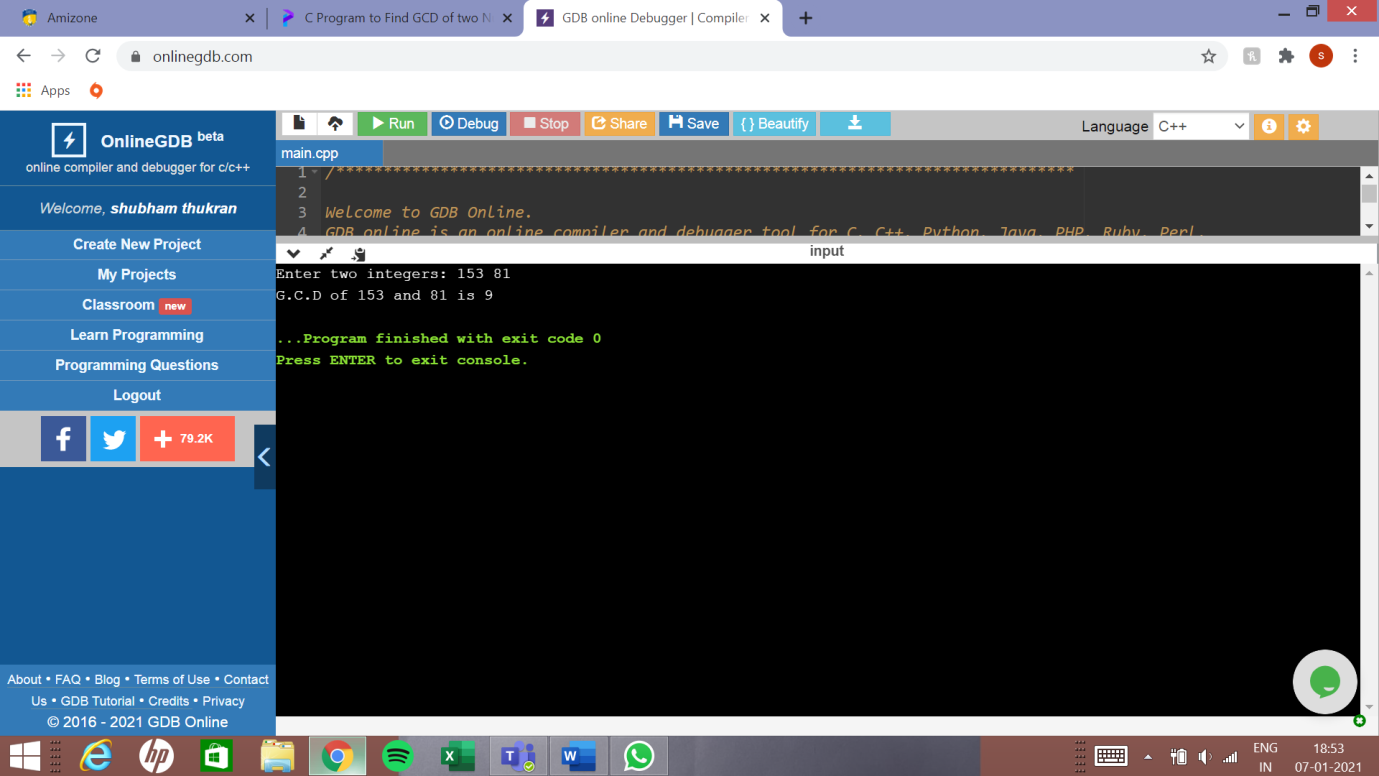
gcd = i;

}

printf("G.C.D of %d and %d is %d", n1, n2, gcd);

return 0;

}



25.Write a program to find LCM of two numbers:

#include <stdio.h>

int main() {

int n1, n2, max;

printf("Enter two positive integers: ");

scanf("%d %d", &n1, &n2);

max = (n1 > n2) ? n1 : n2;

while (1) {

if (max % n1 == 0 && max % n2 == 0) {

printf("The LCM of %d and %d is %d.", n1, n2, max);

break;

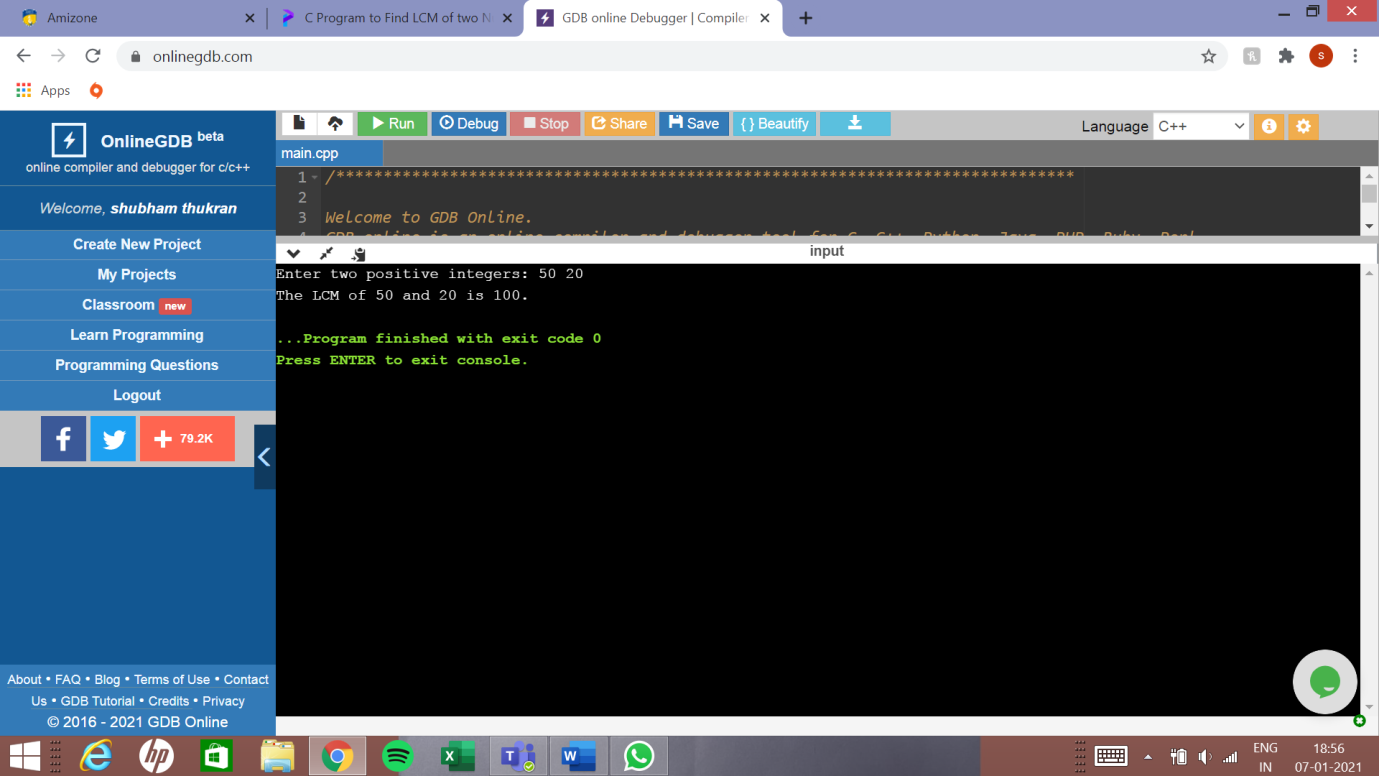
}

max++;

}

return 0;

}



26.Write a program to check whether a number is prime number or not:

#include <stdio.h>

int main() {

int n, i, flag = 0;

printf("Enter a positive integer: ");

scanf("%d", &n);

for (i = 2; i <= n / 2; ++i)

{

if (n % i == 0)

{

flag = 1;

break;

}

}

if (n == 1) {

printf("1 is neither prime nor composite.");

}

else {

if (flag == 0)

printf("%d is a prime number.", n);

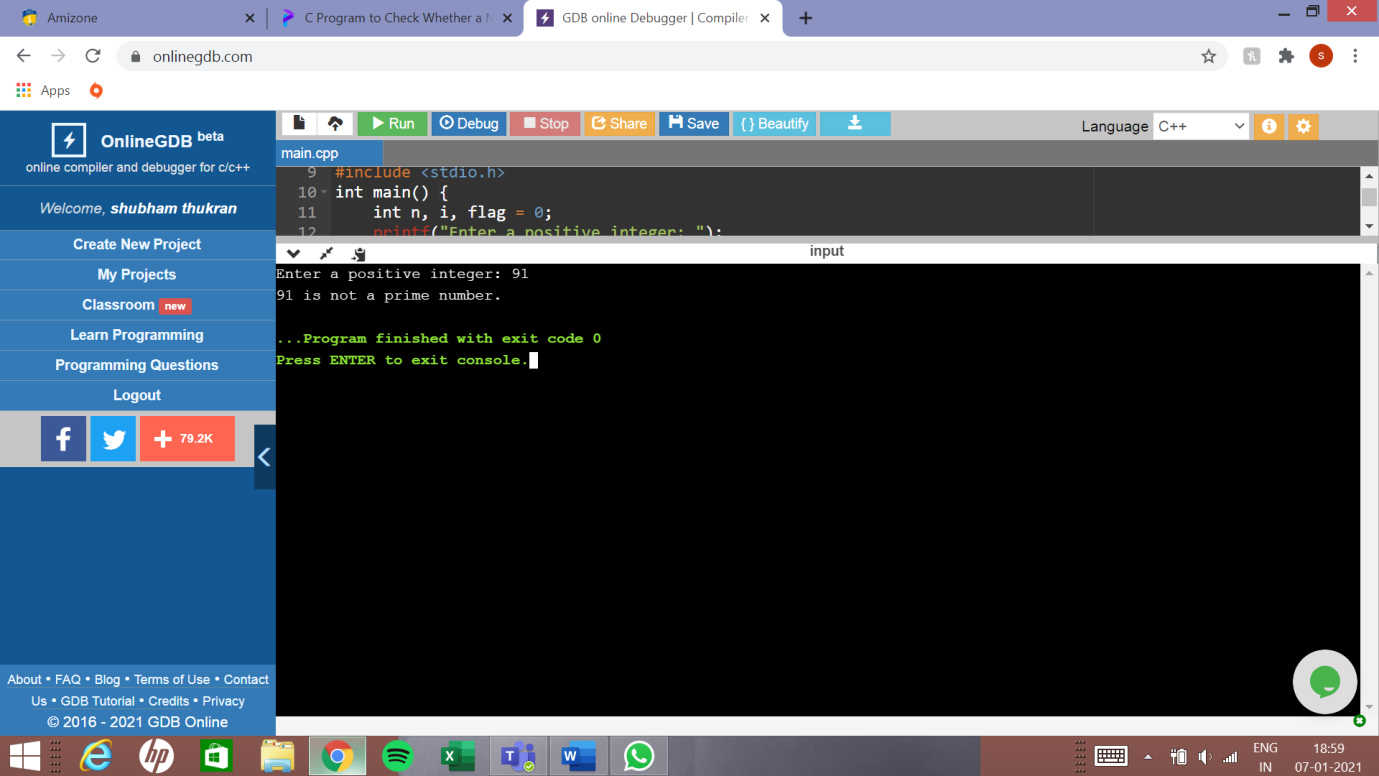
else

printf("%d is not a prime number.", n);

}

return 0;

}



27.Write a program to print all prime numbers between 1 to n:

#include <stdio.h>

int main()

{

int i, j, end, isPrime; // isPrime is used as flag variable

printf("Find prime numbers between 1 to : ");

scanf("%d", &end);

printf("All prime numbers between 1 to %d are:\n", end);

for(i=2; i<=end; i++)

{

isPrime = 1;

for(j=2; j<=i/2; j++)

{

if(i%j==0)

{

isPrime = 0;

break;

}

}

if(isPrime==1)

{

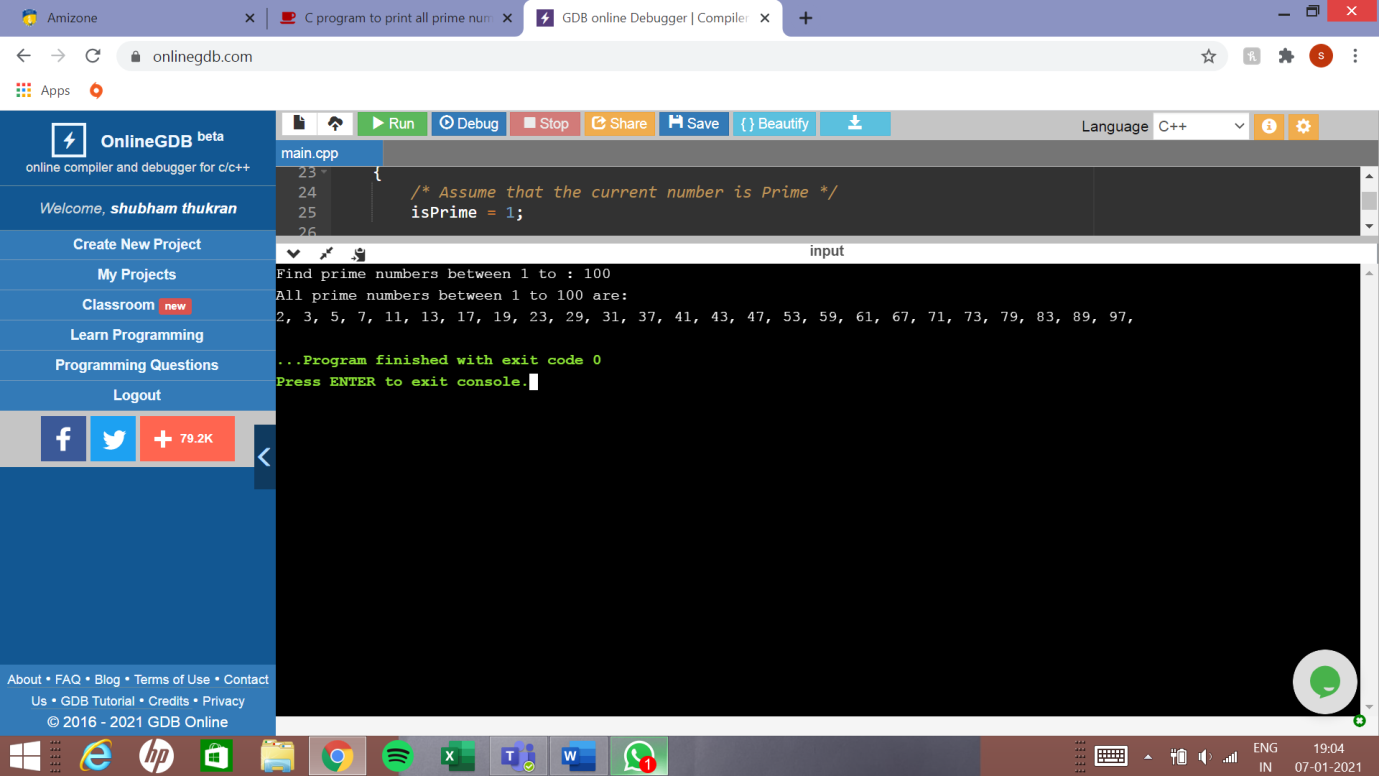
printf("%d, ", i);

}

}

return 0;

}



28. Write a C program to find sum of all prime numbers between 1 to n.

#include <stdio.h>

int main()

{

int i, j, end, isPrime, sum=0;

printf("Find sum of all prime between 1 to : ");

scanf("%d", &end);

for(i=2; i<=end; i++)

{

isPrime = 1;

for(j=2; j<=i/2 ;j++)

{

if(i%j==0)

{

isPrime = 0;

break;

}

}

if(isPrime==1)

{

sum += i;

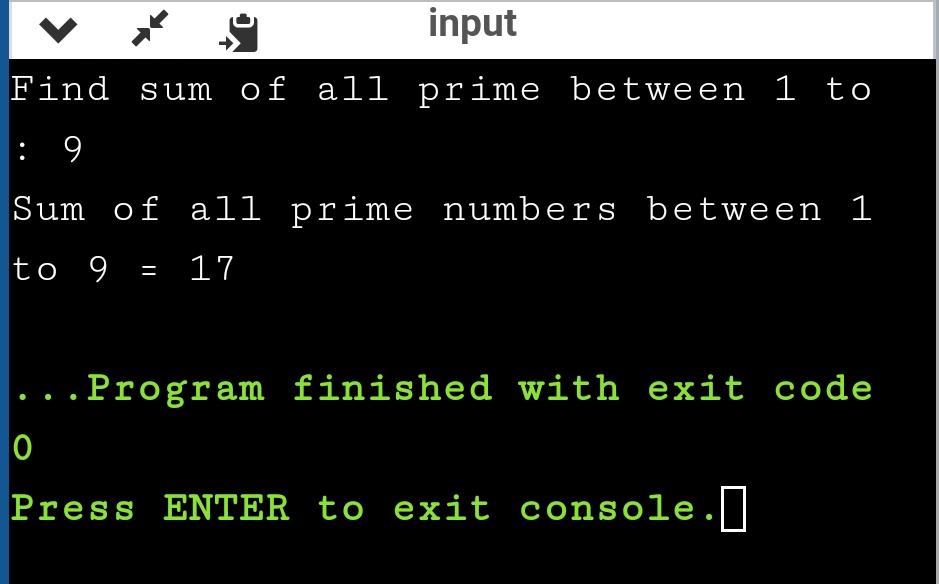
}

}

printf("Sum of all prime numbers between 1 to %d = %d", end, sum);

return 0;

}



29. Write a C program to find all prime factors of a number.

#include <stdio.h>

int main()

{

int i, j, num, isPrime;

printf("Enter any number to print Prime factors: ");

scanf("%d", &num);

printf("All Prime Factors of %d are: \n", num);

for(i=2; i<=num; i++)

{

if(num%i==0)

{

isPrime = 1;

for(j=2; j<=i/2; j++)

{

if(i%j==0)

{

isPrime = 0;

break;

}

}

if(isPrime==1)

{

printf("%d, ", i);

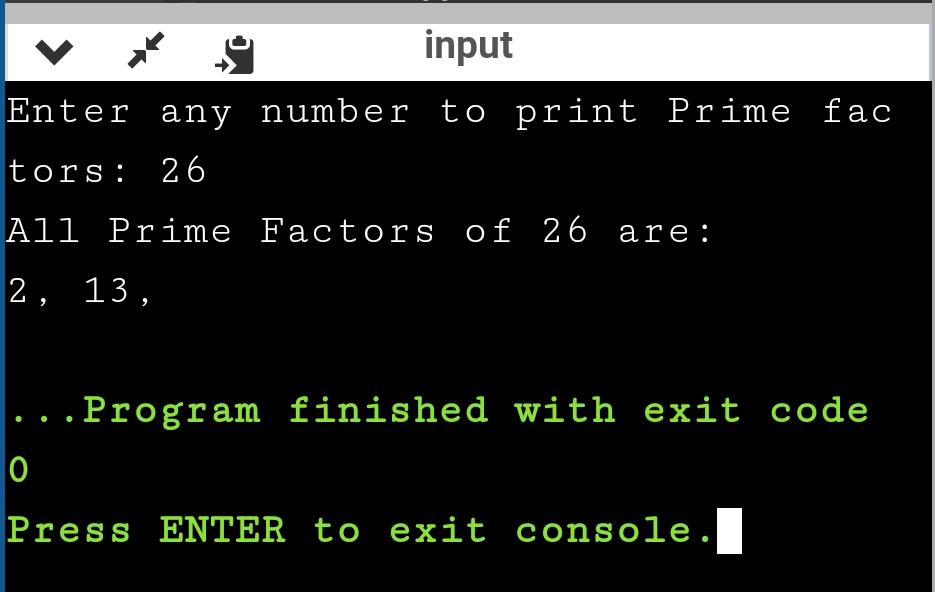
}

}

}

return 0;

}



30. Write a C program to check whether a number is Armstrong number or not.

#include<stdio.h>

int main(){

int num,r,sum=0,temp;

printf("Enter a number: ");

scanf("%d",&num);

temp=num;

while(num!=0){

r=num%10;

num=num/10;

sum=sum+(r\*r\*r);

}

if(sum==temp)

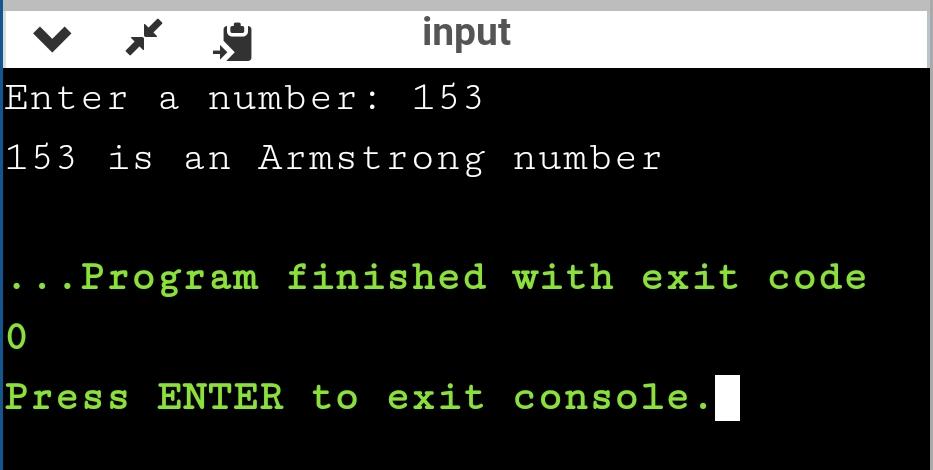
printf("%d is an Armstrong number",temp);

else

printf("%d is not an Armstrong number",temp);

return 0;

}



31. Write a C program to print all Armstrong numbers between 1 to n.

#include <stdio.h>

int checkArmstrong(int num)

{

int tempNumber,rem,sum;

tempNumber=num;

sum=0;

while(tempNumber!=0)

{

rem=tempNumber%10;

sum=sum + (rem\*rem\*rem);

tempNumber/=10;

}

if(sum==num)

return 1;

else

return 0;

}

int main()

{

int i,n;

printf("Enter the value of N: ");

scanf("%d",&n);

printf("All Armstrong numbers from 1 to %d:\n",n);

for(i=1;i<=n;i++)

{

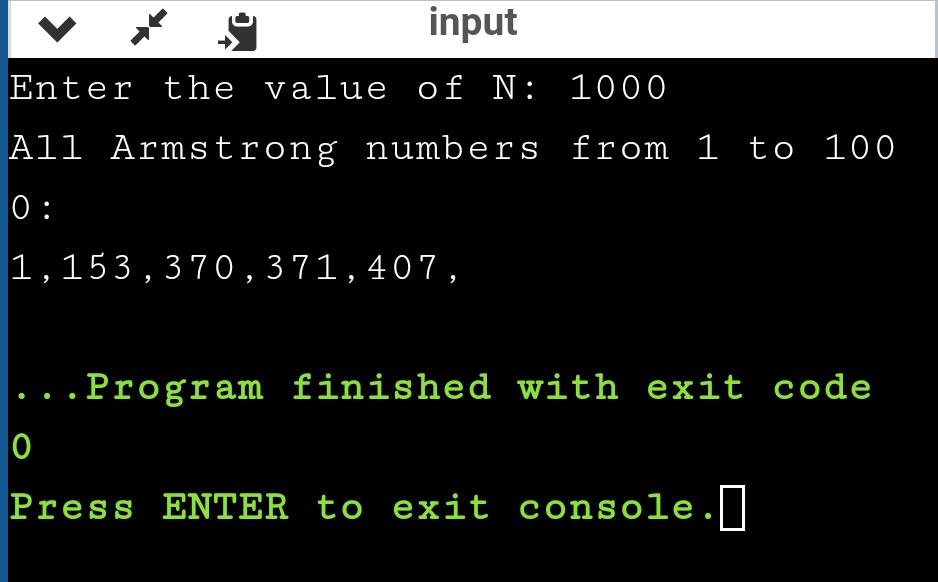
if(checkArmstrong(i))

printf("%d,",i);

}

return 0;

}



32. Write a C program to check whether a number is Perfect number or not.

#include <stdio.h>

int main()

{

int i, num, sum = 0;

printf("Enter any number to check perfect number: ");

scanf("%d", &num);

for(i = 1; i <= num / 2; i++)

{

if(num%i == 0)

{

sum += i;

}

}

if(sum == num)

{

printf("%d is PERFECT NUMBER", num);

}

else

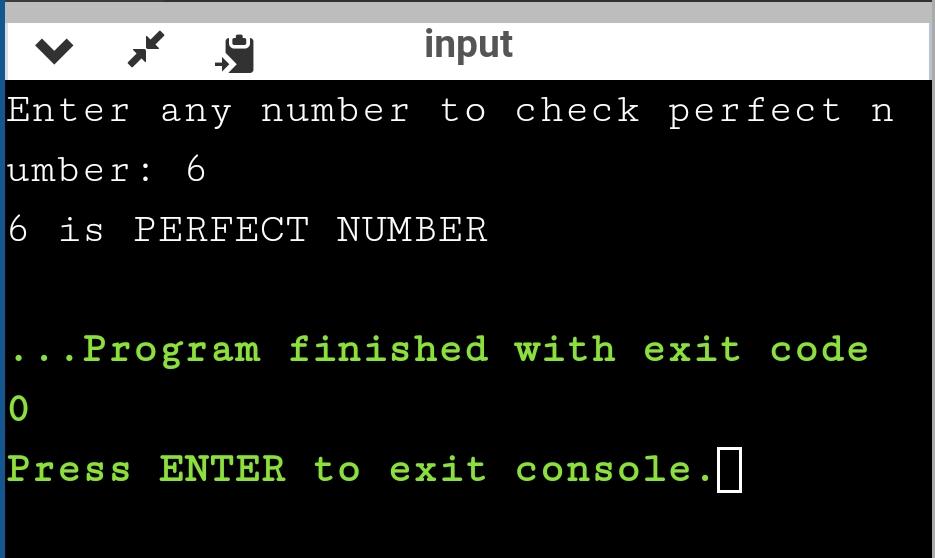
{

printf("%d is NOT PERFECT NUMBER", num);

}

return 0;

}



33. Write a C program to print all Perfect numbers between 1 to n.

#include <stdio.h>

int main()

{

int i, j, end, sum;

printf("Enter upper limit: ");

scanf("%d", &end);

printf("All Perfect numbers between 1 to %d:\n", end);

for(i=1; i<=end; i++)

{

sum = 0;

for(j=1; j<i; j++)

{

if(i % j == 0)

{

sum += j;

}

}

if(sum == i)

{

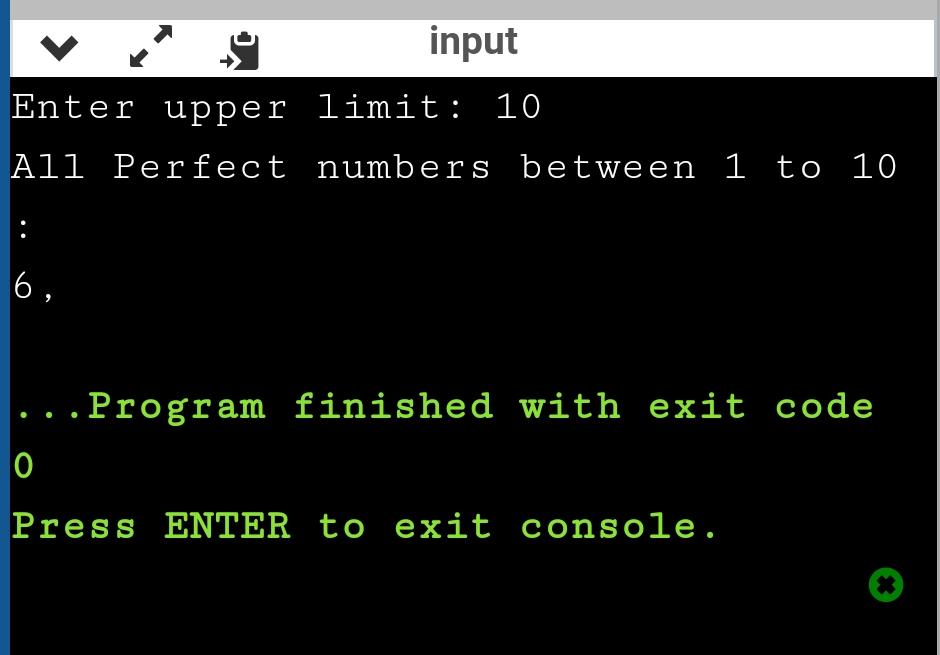
printf("%d, ", i);

}

}

return 0;

}



34. Write a C program to check whether a number is Strong number or not.

#include <stdio.h>

int main()

{

int i, originalNum, num, lastDigit, sum;

long fact;

printf("Enter any number to check Strong number: ");

scanf("%d", &num);

originalNum = num;

sum = 0;

while(num > 0)

{

lastDigit = num % 10;

fact = 1;

for(i=1; i<=lastDigit; i++)

{

fact = fact \* i;

}

sum = sum + fact;

num = num / 10;

}

if(sum == originalNum)

{

printf("%d is STRONG NUMBER", originalNum);

}

else

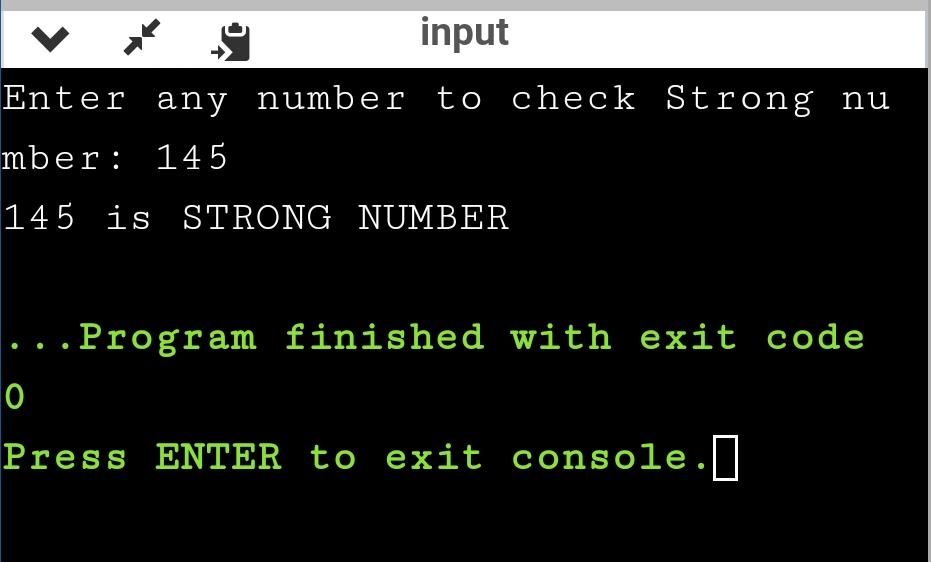
{

printf("%d is NOT STRONG NUMBER", originalNum);

}

return 0;

}



35. Write a C program to print all Strong numbers between 1 to n.

#include <stdio.h>

int main()

{

int i, j, cur, lastDigit, end;

long long fact, sum;

printf("Enter upper limit: ");

scanf("%d", &end);

printf("All Strong numbers between 1 to %d are:\n", end);

for(i=1; i<=end; i++)

{

cur = i;

sum = 0;

while(cur > 0)

{

fact = 1ll;

lastDigit = cur % 10;

for( j=1; j<=lastDigit; j++)

{

fact = fact \* j;

}

sum += fact;

cur /= 10;

}

if(sum == i)

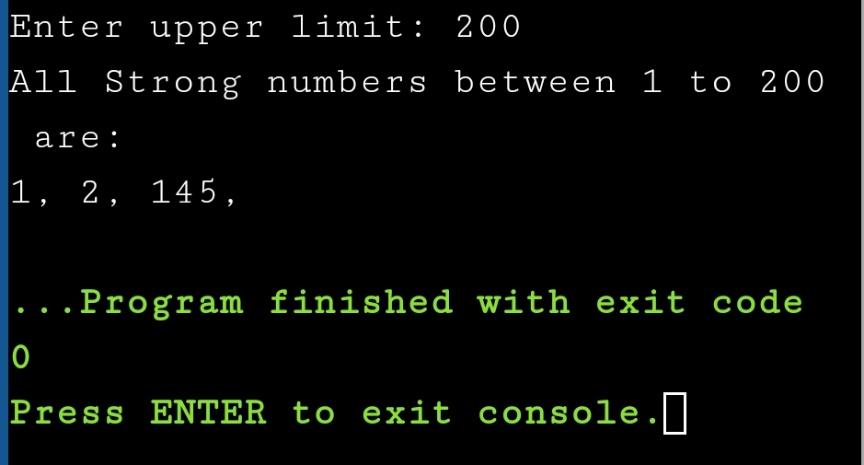
{

printf("%d, ", i);

}

}

return 0;

}

36. Write a C program to print Fibonacci series up to n terms.

#include <stdio.h>

int main()

{

int a, b, c, i, terms;

printf("Enter number of terms: ");

scanf("%d", &terms);

a = 0;

b = 1;

c = 0;

printf("Fibonacci terms: \n");

for(i=1; i<=terms; i++)

{

printf("%d, ", c);

a = b;

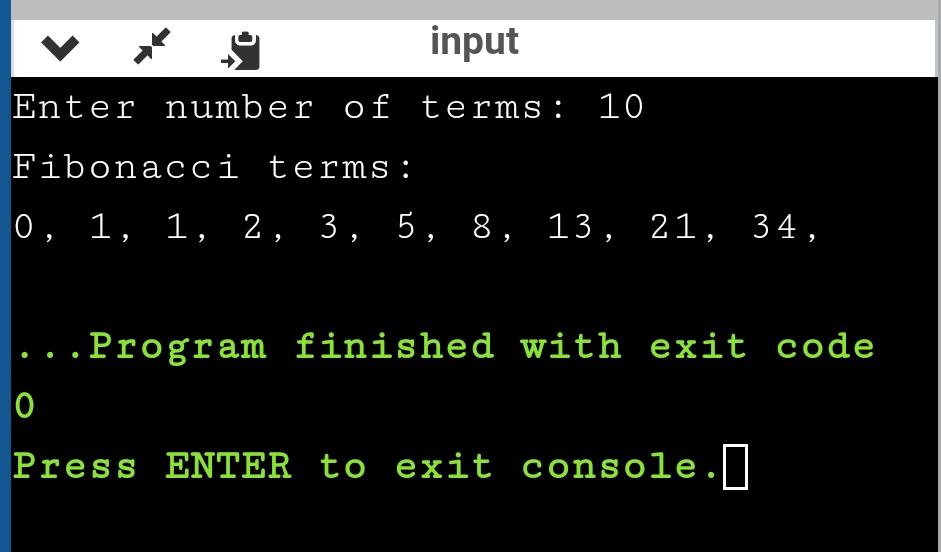
b = c;

c = a + b; // New term

}

return 0;

}



37. Write a C program to find one’s complement of a binary number

#include <stdio.h>

#define SIZE 8

int main()

{

char binary[SIZE + 1], onesComp[SIZE + 1];

int i, error=0;

printf("Enter %d bit binary value: ", SIZE);

scanf("%s",binary);

for(i=0; i<SIZE; i++)

{

if(binary[i] == '1')

{

onesComp[i] = '0';

}

else if(binary[i] == '0')

{

onesComp[i] = '1';

}

else

{

printf("Invalid Input");

error = 1;

/\* Exits from loop \*/

break;

}

}

onesComp[SIZE] = '\0';

if(error == 0)

{

printf("Original binary = %s\n", binary);

printf("Ones complement = %s", onesComp);

}

return 0;

}

Enter 8 bit binary value: 01000011

Original binary = 01000011

Ones complement = 10111100

**...Program finished with exit code 0**

**Press ENTER to exit console.**

38. Write in C program to find two’s complemwnt of a binary number.

#include <stdio.h>

#define SIZE 8

int main()

{

char binary[SIZE + 1], onesComp[SIZE + 1], twosComp[SIZE + 1];

int i, carry=1;

printf("Enter %d bit binary value: ", SIZE);

scanf("%s",binary);

for(i=0; i<SIZE; i++)

{

if(binary[i] == '1')

{

onesComp[i] = '0';

}

else if(binary[i] == '0')

{

onesComp[i] = '1';

}

}

onesComp[SIZE] = '\0';

for(i=SIZE-1; i>=0; i--)

{

if(onesComp[i] == '1' && carry == 1)

{

twosComp[i] = '0';

}

else if(onesComp[i] == '0' && carry == 1)

{

twosComp[i] = '1';

carry = 0;

}

else

{

twosComp[i] = onesComp[i];

}

}

twosComp[SIZE] = '\0';

printf("Original binary = %s\n", binary);

printf("Ones complement = %s\n", onesComp);

printf("Twos complement = %s\n", twosComp);

return 0;

}

Enter 8 bit binary value: 01101100

Original binary = 01101100

Ones complement = 10010011

Twos complement = 10010100

**...Program finished with exit code 0**

**Press ENTER to exit console.**

39.Write a C program to convert binary to octal number system.

#include <stdio.h>

int main()

{

int octalConstant[] = {0, 1, 10, 11, 100, 101, 110, 111};

long long binary, octal, tempBinary;

int digit, place, i;

octal = 0;

place= 1;

printf("Enter any binary number: ");

scanf("%lld", &binary);

tempBinary = binary;

while(tempBinary != 0)

{

digit = tempBinary % 1000;

for(i=0; i<8; i++)

{

if(octalConstant[i] == digit)

{

octal = (i \* place) + octal;

break;

}

}

tempBinary /= 1000;

place \*= 10;

}

printf("Original binary number = %lld\n", binary);

printf("Octal number = %lld", octal);

return 0;

}

Enter any binary number: 00110111

Original binary number = 110111

Octal number = 67

**...Program finished with exit code 0**                                                                                                    **Press ENTER to exit console.**

40.Write a C program to convert binary to decimal number system.

#include <stdio.h>

#include <math.h>

#define BASE 2

int main()

{

long long binary, decimal=0, tempBinary;

int N=0;

printf("Enter any binary number: ");

scanf("%lld", &binary);

tempBinary = binary;

while(tempBinary!=0)

{

if(tempBinary % 10 == 1)

{

decimal += pow(BASE, N);

}

N++;

tempBinary /= 10;

}

printf("Binary number = %lld\n", binary);

printf("Decimal number= %lld", decimal);

return 0;

}

Enter any binary number: 01101110

Binary number = 1101110

Decimal number= 110

**...Program finished with exit code 0**

**Press ENTER to exit console.**

41.Write a C program to convert Binary to Hexadecimal number system.

#include <stdio.h>

#include <string.h>

int main()

{

int hexConstant[] = {0, 1, 10, 11, 100, 101, 110, 111, 1000,

1001, 1010, 1011, 1100, 1101, 1110, 1111};

long long binary, tempBinary;

char hex[20];

int index, i, digit;

printf("Enter binary number: ");

scanf("%lld", &binary);

tempBinary = binary;

index = 0;

while(tempBinary!=0)

{

digit = tempBinary % 10000;

for(i=0; i<16; i++)

{

if(hexConstant[i] == digit)

{

if(i<10)

{

hex[index] = (char)(i + 48);

}

else

{

hex[index] = (char)((i-10) + 65);

}

index++;

break;

}

}

tempBinary /= 10000;

}

hex[index] = '\0';

strrev(hex);

printf("Binary number = %lld\n", binary);

printf("Hexadecimal number = %s", hex);

return 0;

}

Enter binary number: 01101110

Binary number = 1101110

Hexadecimal number = 6E

**...Program finished with exit code 0**

**Press ENTER to exit console.**

42.Write a C program to convert octal to binary number system.

#include <stdio.h>

int main()

{

int OCTALVALUES[] = {0, 1, 10, 11, 100, 101, 110, 111};

long long octal, tempOctal, binary, place;

int rem;

/\* Input Octal number from user \*/

printf("Enter any Octal number: ");

scanf("%lld", &octal);

tempOctal = octal;

binary = 0;

place = 1;

/\* Convert octal to binary \*/

while(tempOctal > 0)

{

/\* Extract the last digit of octal \*/

rem = tempOctal % 10;

/\*

\* Get the binary equivalent of octal digit

\* add it to the binary variable

\*/

binary = (OCTALVALUES[rem] \* place) + binary;

/\* Remove the last octal digit \*/

tempOctal /= 10;

/\* Increase the place value \*/

place \*= 1000;

}

printf("Octal number = %lld\n", octal);

printf("Binary number = %lld", binary);

return 0;

}

Enter any Octal number: 1720

Octal number = 1720

Binary number = 1111010000

**...Program finished with exit code 0**

**Press ENTER to exit console.**

43.Write a C program to convert octal to decimal number system.

#include <stdio.h>

#include <math.h>

int main()

{

long long octal, tempOctal, decimal;

int rem, place;

/\* Input octal number from user \*/

printf("Enter any octal number: ");

scanf("%lld", &octal);

tempOctal = octal;

decimal = 0;

place = 0;

/\*

\* Convert octal to decimal

\*/

while(tempOctal > 0)

{

/\* Extract the last digit of octal \*/

rem = tempOctal % 10;

/\* Convert last octal digit to decimal \*/

decimal += pow(8, place) \* rem;

/\* Remove the last octal digit \*/

tempOctal /= 10;

place++;

}

printf("Octal number = %lld\n", octal);

printf("Decimal number = %lld", decimal);

return 0;

}

Enter any octal number: 172

Octal number = 172

Decimal number = 122

**...Program finished with exit code 0**

**Press ENTER to exit console.**

44.Write a C program to convert octal to hexadecimal to number system.

#include <stdio.h>

int main()

{

int OCTALVALUES[] = {0, 1, 10, 11, 100, 101, 110, 111};

long long octal, tempOctal, binary, place;

char hex[65] = "";

int rem;

place = 1;

binary = 0;

/\* Input octal number from user \*/

printf("Enter any octal number: ");

scanf("%lld", &octal);

tempOctal = octal;

/\*

\* Octal to binary conversion

\*/

while(tempOctal > 0)

{

rem = tempOctal % 10;

binary = (OCTALVALUES[rem] \* place) + binary;

tempOctal /= 10;

place \*= 1000;

}

/\*

\* Binary to hexadecimal conversion

\*/

while(binary > 0)

{

rem = binary % 10000;

switch(rem)

{

case 0:

strcat(hex, "0");

break;

case 1:

strcat(hex, "1");

break;

case 10:

strcat(hex, "2");

break;

case 11:

strcat(hex, "3");

break;

case 100:

strcat(hex, "4");

break;

case 101:

strcat(hex, "5");

break;

case 110:

strcat(hex, "6");

break;

case 111:

strcat(hex, "7");

break;

case 1000:

strcat(hex, "8");

break;

case 1001:

strcat(hex, "9");

break;

case 1010:

strcat(hex, "A");

break;

case 1011:

strcat(hex, "B");

break;

case 1100:

strcat(hex, "C");

break;

case 1101:

strcat(hex, "D");

break;

case 1110:

strcat(hex, "E");

break;

case 1111:

strcat(hex, "F");

break;

}

binary /= 10000;

}

strrev(hex);

printf("Octal number: %lld\n", octal);

printf("Hexadecimal number: %s", hex);

return 0;

}

Enter any octal number: 125715

Octal number: 125715

Hexadecimal number: ABCD

**...Program finished with exit code 0**

**Press ENTER to exit console.**

45.Write a C program to convert decimal to binary number system.

#include <stdio.h>

int main()

{

long long decimal, tempDecimal, binary;

int rem, place = 1;

binary = 0;

printf("Enter any decimal number: ");

scanf("%lld", &decimal);

tempDecimal = decimal;

while(tempDecimal > 0)

{

rem = tempDecimal % 2;

binary = (rem \* place) + binary;

tempDecimal /= 2;

place \*= 10;

}

printf("Decimal number = %lld\n", decimal);

printf("Binary number = %lld", binary);

return 0;

}

Enter any decimal number: 112

Decimal number = 112

Binary number = 1110000

**...Program finished with exit code 0**

**Press ENTER to exit console.**

46. Write a C program to convert decimal to octal number system.

#include <stdio.h>

int main()

{

long long decimal, tempDecimal, octal;

int i, rem, place = 1;

octal = 0;

/\* Input decimal number from user \*/

printf("Enter any decimal number: ");

scanf("%lld", &decimal);

tempDecimal = decimal;

/\* Decimal to octal conversion \*/

while(tempDecimal > 0)

{

rem = tempDecimal % 8;

octal = (rem \* place) + octal;

tempDecimal /= 8;

place \*= 10;

}

printf("\nDecimal number = %lld\n", decimal);

printf("Octal number = %lld", octal);

return 0;

}

Enter any decimal number: 20

Decimal number = 20

Octal number = 24

**...Program finished with exit code 0**

**Press ENTER to exit console.**

47. Write a C program to convert decimal to hexadecimal .

#include<stdio.h>

#include<conio.h>

int main()

{

int decnum, rem, i=0;

char hexnum[50];

printf("Enter any decimal number: ");

scanf("%d", &decnum);

while(decnum!=0)

{

rem = decnum%16;

if(rem<10)

rem = rem+48;

else

rem = rem+55;

hexnum[i] = rem;

i++;

decnum = decnum/16;

}

printf("\nEquivalent Value in Hexadecimal = ");

for(i=i-1; i>=0; i--)

printf("%c", hexnum[i]);

getch();

return 0;

}



48.Write a C program to convert hexadecimal to binary number system.

#include <stdio.h>

#include <string.h>

int main()

{

char hex[17], bin[65] = "";

int i = 0;

/\* Input hexadecimal number from user \*/

printf("Enter any hexadecimal number: ");

gets(hex);

/\* Extract first digit and find binary of each hex digit \*/

for(i=0; hex[i]!='\0'; i++)

{

switch(hex[i])

{

case '0':

strcat(bin, "0000");

break;

case '1':

strcat(bin, "0001");

break;

case '2':

strcat(bin, "0010");

break;

case '3':

strcat(bin, "0011");

break;

case '4':

strcat(bin, "0100");

break;

case '5':

strcat(bin, "0101");

break;

case '6':

strcat(bin, "0110");

break;

case '7':

strcat(bin, "0111");

break;

case '8':

strcat(bin, "1000");

break;

case '9':

strcat(bin, "1001");

break;

case 'a':

case 'A':

strcat(bin, "1010");

break;

case 'b':

case 'B':

strcat(bin, "1011");

break;

case 'c':

case 'C':

strcat(bin, "1100");

break;

case 'd':

case 'D':

strcat(bin, "1101");

break;

case 'e':

case 'E':

strcat(bin, "1110");

break;

case 'f':

case 'F':

strcat(bin, "1111");

break;

default:

printf("Invalid hexadecimal input.");

}

}

printf("Hexademial number = %s\n", hex);

printf("Binary number = %s", bin);

return 0;

}

Enter any hexadecimal number: 1a

Hexademial number = 1a

Binary number = 00011010

**...Program finished with exit code 0**

**Press ENTER to exit console.**

49.Write a C program to convert hexadecimal to octal number system.

#include <stdio.h>

int main()

{

char hex[17];

long long octal, bin, place;

int i = 0, rem, val;

/\* Input hexadecimal number from user \*/

printf("Enter any hexadecimal number: ");

gets(hex);

octal = 0ll;

bin = 0ll;

place = 0ll;

/\* Hexadecimal to binary conversion \*/

for(i=0; hex[i]!='\0'; i++)

{

bin = bin \* place;

switch(hex[i])

{

case '0':

bin += 0;

break;

case '1':

bin += 1;

break;

case '2':

bin += 10;

break;

case '3':

bin += 11;

break;

case '4':

bin += 100;

break;

case '5':

bin += 101;

break;

case '6':

bin += 110;

break;

case '7':

bin += 111;

break;

case '8':

bin += 1000;

break;

case '9':

bin += 1001;

break;

case 'a':

case 'A':

bin += 1010;

break;

case 'b':

case 'B':

bin += 1011;

break;

case 'c':

case 'C':

bin += 1100;

break;

case 'd':

case 'D':

bin += 1101;

break;

case 'e':

case 'E':

bin += 1110;

break;

case 'f':

case 'F':

bin += 1111;

break;

default:

printf("Invalid hexadecimal input.");

}

place = 10000;

}

place = 1;

/\* Binary to octal conversion \*/

while(bin > 0)

{

rem = bin % 1000;

switch(rem)

{

case 0:

val = 0;

break;

case 1:

val = 1;

break;

case 10:

val = 2;

break;

case 11:

val = 3;

break;

case 100:

val = 4;

break;

case 101:

val = 5;

break;

case 110:

val = 6;

break;

case 111:

val = 7;

break;

}

octal = (val \* place) + octal;

bin /= 1000;

place \*= 10;

}

printf("Hexadecimal number = %s\n", hex);

printf("Octal number = %lld", octal);

return 0;

}

Enter any hexadecimal number: fff

Hexadecimal number = fff

Octal number = 7777

**...Program finished with exit code 0**

**Press ENTER to exit console.**

50. Write a C program to convert hexadecimal to decimal number system.

#include <stdio.h>

#include <math.h>

#include <string.h>

int main()

{

char hex[17];

long long decimal, place;

int i = 0, val, len;

decimal = 0;

place = 1;

/\* Input hexadecimal number from user \*/

printf("Enter any hexadecimal number: ");

gets(hex);

/\* Find the length of total number of hex digit \*/

len = strlen(hex);

len--;

/\*

\* Iterate over each hex digit

\*/

for(i=0; hex[i]!='\0'; i++)

{

/\* Find the decimal representation of hex[i] \*/

if(hex[i]>='0' && hex[i]<='9')

{

val = hex[i] - 48;

}

else if(hex[i]>='a' && hex[i]<='f')

{

val = hex[i] - 97 + 10;

}

else if(hex[i]>='A' && hex[i]<='F')

{

val = hex[i] - 65 + 10;

}

decimal += val \* pow(16, len);

len--;

}

printf("Hexadecimal number = %s\n", hex);

printf("Decimal number = %lld", decimal);

return 0;

}

Enter any hexadecimal number: 1a

Hexadecimal number = 1a

Decimal number = 26

**...Program finished with exit code 0**

**Press ENTER to exit console.**

PATTERN EXERCISE

1)Star pattern programs. write a c program to print the given star pattern.

a)Pyramid star pattern

#include <stdio.h>

int main() {

int i, space, rows, k = 0;

printf("Enter the number of rows: ");

scanf("%d", &rows);

for (i = 1; i <= rows; ++i, k = 0) {

for (space = 1; space <= rows - i; ++space) {

printf(" ");

}

while (k != 2 \* i - 1) {

printf("\* ");

++k;

}

printf("\n");

}

return 0;

}



b.)Hollow pyramid

#include <stdio.h>

int main()

{

int i, j, rows;

/\* Input rows to print from user \*/

printf("Enter number of rows : ");

scanf("%d", &rows);

for(i=1; i<=rows; i++)

{

/\* Print trailing spaces \*/

for(j=i; j<rows; j++)

{

printf(" ");

}

/\* Print hollow pyramid \*/

for(j=1; j<=(2\*i-1); j++)

{

/\*

\* Print star for last row (i==rows),

\* first column(j==1) and for

\* last column (j==(2\*i-1)).

\*/

if(i==rows || j==1 || j==(2\*i-1))

{

printf("\*");

}

else

{

printf(" ");

}

}

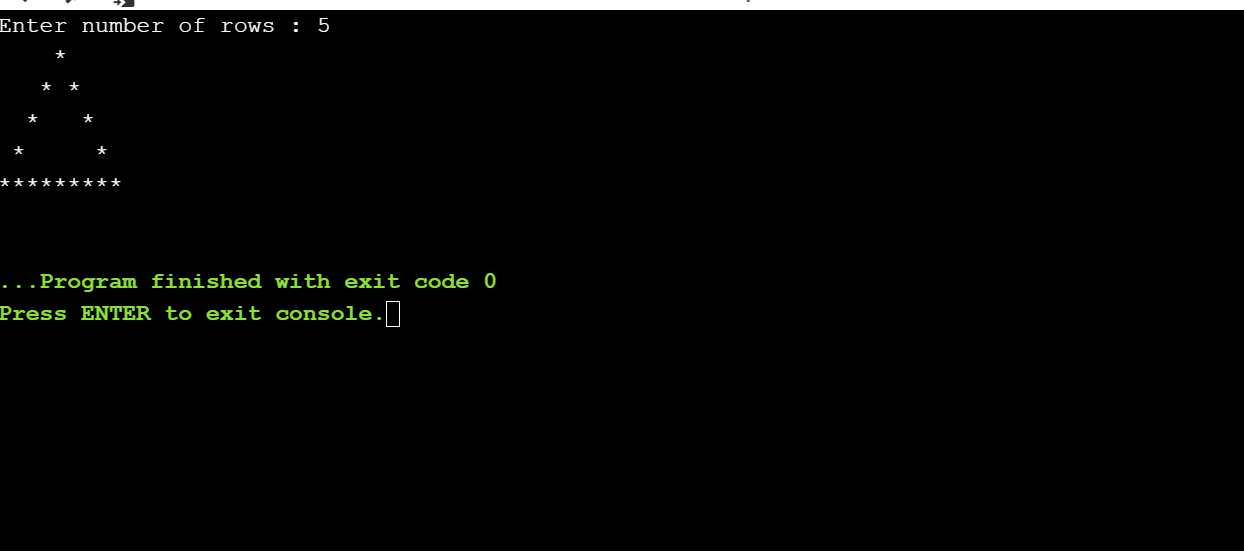
/\* Move to next line \*/

printf("\n");

}

return 0;

}



c.)Inverted pyramid

#include <stdio.h>

int main() {

int rows, i, j, space;

printf("Enter the number of rows: ");

scanf("%d", &rows);

for (i = rows; i >= 1; --i) {

for (space = 0; space < rows - i; ++space)

printf(" ");

for (j = i; j <= 2 \* i - 1; ++j)

printf("\* ");

for (j = 0; j < i - 1; ++j)

printf("\* ");

printf("\n");

}

return 0;

}



d)Hollow inverted pyramid star pattern

#include <stdio.h>

int main()

{

int i, j, rows;

/\* Input rows to print from user \*/

printf("Enter number of rows: ");

scanf("%d", &rows);

/\* Iterate through rows \*/

for(i=1; i<=rows; i++)

{

/\* Print leading spaces \*/

for(j=1; j<i; j++)

{

printf(" ");

}

/\* Print hollow pyramid \*/

for(j=1; j<=(rows\*2 - (2\*i-1)); j++)

{

/\*

\* Print star for first row(i==1),

\* ith column (j==1) and for

\* last column (rows\*2-(2\*i-1))

\*/

if(i==1 || j==1 || j==(rows\*2 - (2\*i - 1)))

{

printf("\*");

}

else

{

printf(" ");

}

}

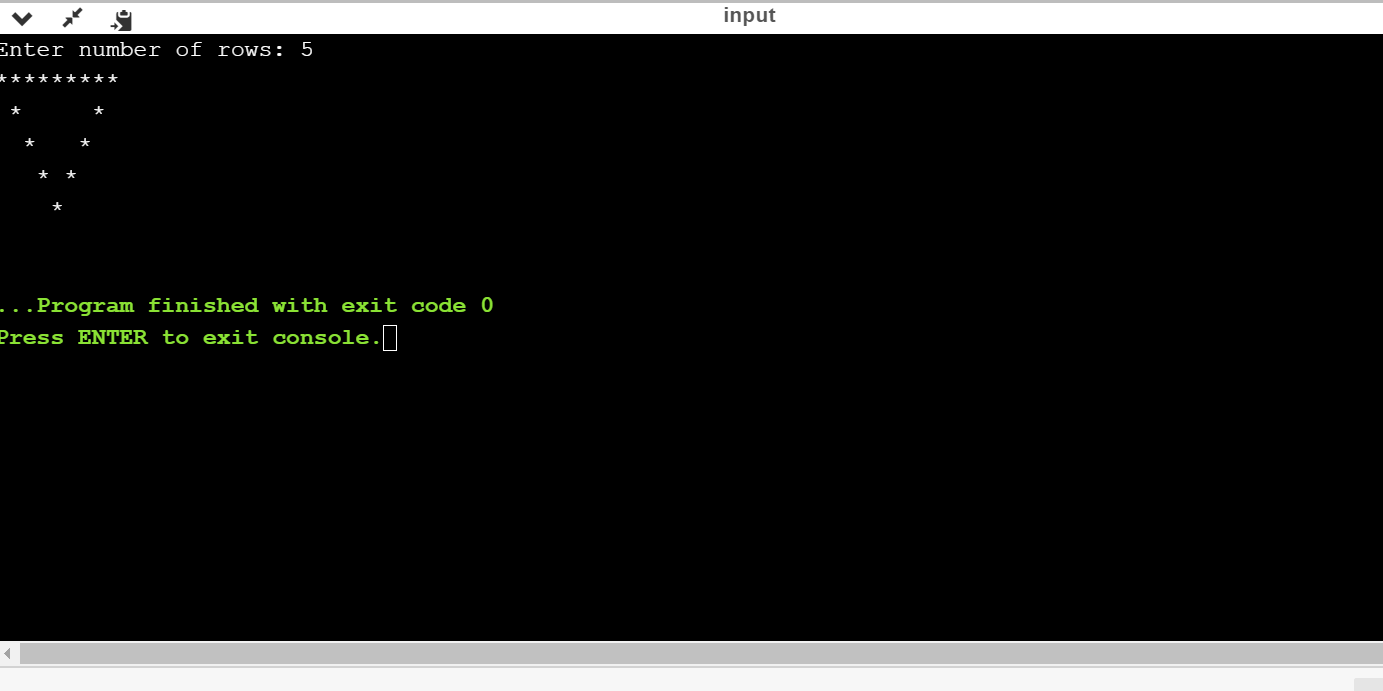
/\* Move to next line \*/

printf("\n");

}

return 0;

}



e)Half diamond star pattern

int i, j, N, columns;

/\* Input number of columns from user \*/

printf("Enter number of columns:");

scanf("%d",&N);

columns=1;

for(i=1;i<N\*2;i++)

{

for(j=1; j<=columns; j++)

{

printf("\*");

}

if(i < N)

{

/\* Increment number of columns per row for upper part \*/

columns++;

}

else

{

/\* Decrement number of columns per row for lower part \*/

columns--;

}

/\* Move to next line \*/

printf("\n");

}

return 0;

}

Enter number of columns:5

\*

\*\*

\*\*\*

\*\*\*\*

\*\*\*\*\*

\*\*\*\*

\*\*\*

\*\*

\*

**...Program finished with exit code 0**

**Press ENTER to exit console.**

f)Mirrored half diamond star pattern

#include <stdio.h>

int main()

{

int i, j, N;

int star, spaces;

/\* Input number of columns to print from user \*/

printf("Enter number of columns : ");

scanf("%d", &N);

spaces = N-1;

star = 1;

/\* Iterate through rows \*/

for(i=1; i<N\*2; i++)

{

/\* Print leading spaces \*/

for(j=1; j<=spaces; j++)

printf(" ");

/\* Print stars \*/

for(j=1; j<=star; j++)

printf("\*");

/\* Move to next line \*/

printf("\n");

if(i < N)

{

star++;

spaces--;

}

else

{

star--;

spaces++;

}

}

return 0;

}

Enter number of columns : 5

    \*

   \*\*

  \*\*\*

 \*\*\*\*

\*\*\*\*\*

 \*\*\*\*

  \*\*\*

   \*\*

    \*

**...Program finished with exit code 0**

**Press ENTER to exit console.**

2)Number pattern programs-write a c program to print the given number of patterns

a)Square number patterns

#include <stdio.h>

int main()

{

int rows, cols, i, j;

/\* Input rows and columns from user \*/

printf("Enter number of rows: ");

scanf("%d", &rows);

printf("Enter number of columns: ");

scanf("%d", &cols);

/\* Iterate through rows \*/

for(i=1; i<=rows; i++)

{

/\* Iterate through columns \*/

for(j=1; j<=cols; j++)

{

printf("1");

}

printf("\n");

}

return 0;

}

Enter number of rows: 5

Enter number of columns: 5

11111

11111

11111

11111

11111

**...Program finished with exit code 0**

**Press ENTER to exit console.**

b)Number pattern 1

#include <stdio.h>

int main()

{

int rows, cols, i, j;

/\* Input rows and columns from user \*/

printf("Enter number of rows: ");

scanf("%d", &rows);

printf("Enter number of columns: ");

scanf("%d", &cols);

for(i=1; i<=rows; i++)

{

for(j=1; j<=cols; j++)

{

// Print 1 if current row is odd

if(i%2 == 1)

{

printf("1");

}

else

{

printf("0");

}

}

printf("\n");

}

return 0;

}

Enter number of rows: 5

Enter number of columns: 5

11111

00000

11111

00000

11111

**...Program finished with exit code 0**

**Press ENTER to exit console.**

c) Number pattern 2

#include <stdio.h>

int main()

{

int rows, cols, i, j;

/\* Input rows and columns from user \*/

printf("Enter number of rows: ");

scanf("%d", &rows);

printf("Enter number of columns: ");

scanf("%d", &cols);

for(i=1; i<=rows; i++)

{

for(j=1; j<=cols; j++)

{

// Print 1 if current column is even

if(j%2 == 1)

{

printf("0");

}

else

{

printf("1");

}

}

printf("\n");

}

return 0;

}

Enter number of rows: 5

Enter number of columns: 5

01010

01010

01010

01010

01010

**...Program finished with exit code 0**

**Press ENTER to exit console.**

d)Number pattern 3

#include <stdio.h>

int main()

{

int rows, cols, i, j;

/\* Input rows and columns from user \*/

printf("Enter number of rows: ");

scanf("%d", &rows);

printf("Enter number of columns: ");

scanf("%d", &cols);

for(i=1; i<=rows; i++)

{

for(j=1; j<=cols; j++)

{

/\*

\* Print 1 if its first or last row

\* Print 1 if its first or last column

\*/

if(i==1 || i==rows || j==1 || j==cols)

{

printf("1");

}

else

{

printf("0");

}

}

printf("\n");

}

return 0;

}

Enter number of rows: 5

Enter number of columns: 5

11111

10001

10001

10001

11111

**...Program finished with exit code 0**

**Press ENTER to exit console.**

e) Number pattern 4

#include <stdio.h>

int main()

{

int rows, cols, i, j;

int centerRow, centerCol;

/\* Input rows and columns from user \*/

printf("Enter number of rows: ");

scanf("%d", &rows);

printf("Enter number of columns: ");

scanf("%d", &cols);

/\* Find center row and column \*/

centerRow = (rows + 1) / 2;

centerCol = (cols + 1) / 2;

for(i=1; i<=rows; i++)

{

for(j=1; j<=cols; j++)

{

if(centerCol == j && centerRow == i)

{

printf("0");

}

else if(cols%2 == 0 && centerCol+1 == j)

{

if(centerRow == i || (rows%2 == 0 && centerRow+1 == i))

printf("0");

else

printf("1");

}

else if(rows%2 == 0 && centerRow+1 == i)

{

if(centerCol == j || (cols%2 == 0 && centerCol+1 == j))

printf("0");

else

printf("1");

}

else

{

printf("1");

}

}

printf("\n");

}

return 0;

}

Enter number of rows: 5

Enter number of columns: 5

11111

11111

11011

11111

11111

**...Program finished with exit code 0**

**Press ENTER to exit console.**

f) Number pattern 5

#include <stdio.h>

int main()

{

int rows, cols, i, j, k;

/\* Input rows and columns from user \*/

printf("Enter number of rows: ");

scanf("%d", &rows);

printf("Enter number of columns: ");

scanf("%d", &cols);

k = 1;

for(i=1; i<=rows; i++)

{

for(j=1; j<=cols; j++)

{

if(k == 1)

{

printf("1");

}

else

{

printf("0");

}

// If k = 1 then k \*= -1 => -1

// If k = -1 then k \*= -1 => 1

k \*= -1;

}

if(cols % 2 == 0)

{

k \*= -1;

}

printf("\n");

}

return 0;

}

Enter number of rows: 5

Enter number of columns: 5

10101

01010

10101

01010

10101

**...Program finished with exit code 0**

**Press ENTER to exit console.**

*If…Else Exercises*

1)Write a program to find maximum between two numbers

#include <stdio.h>

int main()

{

int num1,num2;

printf("Enter number 1: ");

scanf("%d",&num1);

printf("Enter number 2: ");

scanf("%d",&num2);

if(num1>num2)

{

printf("Number 1 ---> %d is maximum",num1);

}

else

{

printf("Number 2 ---> %d is maximum",num2);

}

return 0;

}

Enter number 1: 56

Enter number 2: 79

Number 2 ---> 79 is maximum

**...Program finished with exit code 0**

**Press ENTER to exit console.**

2) Write a C program to find maximum between three numbers.

#include <stdio.h>

int main() {

int n1, n2, n3;

printf("Enter three different numbers: ");

scanf("%d %d %d", &n1, &n2, &n3);

if (n1 >= n2 && n1 >= n3)

{

printf("%d is the largest number.", n1);

}

if (n2 >= n1 && n2 >= n3)

{

printf("%d is the largest number.", n2);

}

if (n3 >= n1 && n3 >= n2)

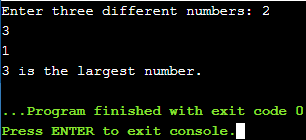
{

printf("%d is the largest number.", n3);

}

return 0;

}



3) Write a C program to check whether a number is negative, positive or zero.

#include <stdio.h>

int main() {

int num;

printf("Enter a number: ");

scanf("%d", &num);

if (num <= 0) {

if (num == 0)

printf("You entered 0.");

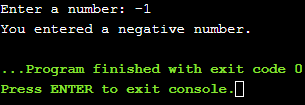
else

printf("You entered a negative number.");

} else

printf("You entered a positive number.");

return 0;}



4) Write a C program to check whether a number is divisible by 5 and 11 or not.

#include <stdio.h>

int main()

{

int num;

printf("Enter any number: ");

scanf("%d", &num);

if((num % 5 == 0) && (num % 11 == 0))

{

printf("Number is divisible by 5 and 11");

}

else

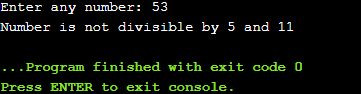
{

printf("Number is not divisible by 5 and 11");

}

return 0;

}



5) Write a C program to check whether a number is even or odd.

#include <stdio.h>

int main() {

int num;

printf("Enter an integer : ");

scanf("%d", &num);

if(num % 2 == 0)

{

printf("%d is even.", num);

}

else

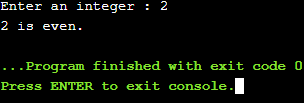
{

printf("%d is odd.", num);

}

return 0;

}



6) Write a C program to check whether a year is leap year or not.

#include <stdio.h>

int main() {

int year;

printf("Enter a year: ");

scanf("%d", &year);

if (year % 400 == 0) {

printf("%d is a leap year.", year);

}

else if (year % 100 == 0) {

printf("%d is not a leap year.", year);

}

else if (year % 4 == 0) {

printf("%d is a leap year.", year);

}

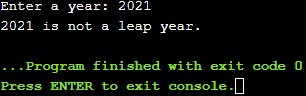
else {

printf("%d is not a leap year.", year);

}

return 0;

}



7) Write a C program to check whether a character is alphabet or not

#include <stdio.h>

int main()

{

char ch;

printf("Enter any character: ");

scanf("%c", &ch);

if((ch >= 'a' && ch <= 'z') || (ch >= 'A' && ch <= 'Z'))

{

printf("'%c' is alphabet.", ch);

}

else if(ch >= '0' && ch <= '9')

{

printf("'%c' is digit.", ch);

}

else

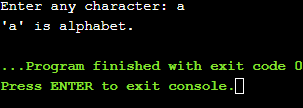
{

printf("'%c' is special character.", ch);

}

return 0;

}



8) Write a C program to input any alphabet and check whether it is vowel or consonant

#include <stdio.h>

int main()

{

char ch;

printf("Enter any character: ");

scanf("%c", &ch);

if(ch=='a' || ch=='e' || ch=='i' || ch=='o' || ch=='u' ||

ch=='A' || ch=='E' || ch=='I' || ch=='O' || ch=='U')

{

printf("'%c' is Vowel.", ch);

}

else if((ch >= 'a' && ch <= 'z') || (ch >= 'A' && ch <= 'Z'))

{

printf("'%c' is Consonant.", ch);

}

else

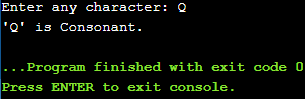
{

printf("'%c' is not an alphabet.", ch);

}

return 0;

}



9) Write a C program to input any character and check whether it is alphabet, digit or special character.

#include <stdio.h>

int main()

{

char ch;

printf("Enter any character: ");

scanf("%c", &ch);

if((ch >= 'a' && ch <= 'z') || (ch >= 'A' && ch <= 'Z'))

{

printf("'%c' is alphabet.", ch);

}

else if(ch >= '0' && ch <= '9')

{

printf("'%c' is digit.", ch);

}

else

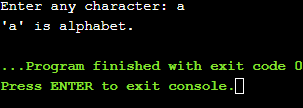
{

printf("'%c' is special character.", ch);

}

return 0;

}



10) Write a C program to check whether a character is uppercase or lowercase alphabet

#include <stdio.h>

int main()

{

char ch;

printf("Enter any character: ");

scanf("%c", &ch);

if(ch >= 'A' && ch <= 'Z')

{

printf("'%c' is uppercase alphabet.", ch);

}

else if(ch >= 'a' && ch <= 'z')

{

printf("'%c' is lowercase alphabet.", ch);

}

else

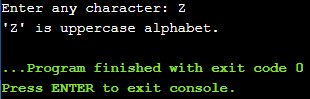
{

printf("'%c' is not an alphabet.", ch);

}

return 0;

}



11): WAP to input week numbers and print week day.

#include <stdio.h>

int main()

{

int week;

printf("Enter week number (1-7): ");

scanf("%d", &week);

if(week == 1)

{

printf("Monday");

}

else if(week == 2)

{

printf("Tuesday");

}

else if(week == 3)

{

printf("Wednesday");

}

else if(week == 4)

{

printf("Thursday");

}

else if(week == 5)

{

printf("Friday");

}

else if(week == 6)

{

printf("Saturday");

}

else if(week == 7)

{

printf("Sunday");

}

else

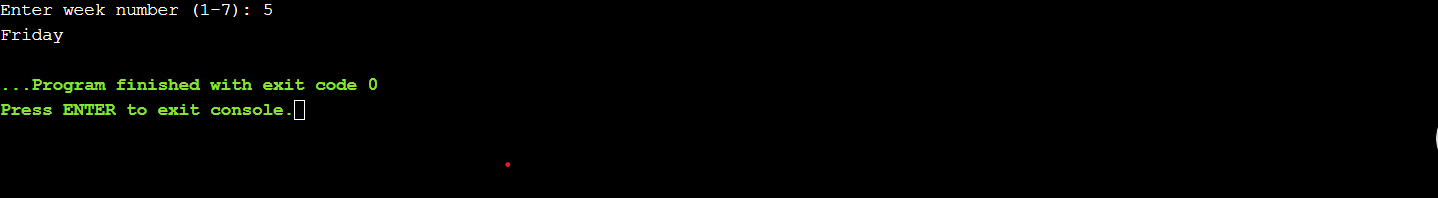
{

printf("Invalid Input! Please enter week number between 1-7.");

}

return 0;

}



12): WAP to input month number and print number of days in the month.

#include <stdio.h>

int main()

{

int month;

printf("Enter month number (1-12): ");

scanf("%d", &month);

if(month == 1 || month==3 || month==5 || month==7 || month==8 || month==10 || month==12)

{

printf("31 days");

}

else if(month == 2)

{

printf("28 or 29 days");

}

else if(month == 4 || month==6 || month==9 || month==11)

{

printf("30 days");

}

else

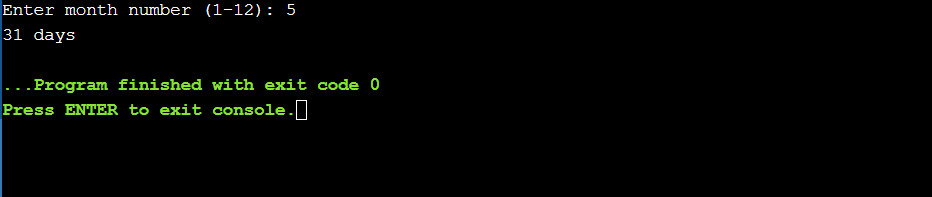
{

printf("Invalid input! Please enter month number between (1-12).");

}

return 0;

}



13 ): WAP to count total numbers of notes in given amount.

#include <stdio.h>

int main()

{

int amount;

int note500, note100, note50, note20, note10, note5, note2, note1;

note500 = note100 = note50 = note20 = note10 = note5 = note2 = note1 = 0;

printf("Enter amount: ");

scanf("%d", &amount);

if(amount >= 500)

{

note500 = amount/500;

amount -= note500 \* 500;

}

if(amount >= 100)

{

note100 = amount/100;

amount -= note100 \* 100;

}

if(amount >= 50)

{

note50 = amount/50;

amount -= note50 \* 50;

}

if(amount >= 20)

{

note20 = amount/20;

amount -= note20 \* 20;

}

if(amount >= 10)

{

note10 = amount/10;

amount -= note10 \* 10;

}

if(amount >= 5)

{

note5 = amount/5;

amount -= note5 \* 5;

}

if(amount >= 2)

{

note2 = amount /2;

amount -= note2 \* 2;

}

if(amount >= 1)

{

note1 = amount;

}

printf("Total number of notes = \n");

printf("500 = %d\n", note500);

printf("100 = %d\n", note100);

printf("50 = %d\n", note50);

printf("20 = %d\n", note20);

printf("10 = %d\n", note10);

printf("5 = %d\n", note5);

printf("2 = %d\n", note2);

printf("1 = %d\n", note1);

return 0;

}



**\*\*\*\*\*\*\*\*PROGRAM 14 AND PROGRAM 15 ARE SAME AS PROGRAM 12 AND PROGRAM 13.\*\*\*\*\*\*\*\***

16): WAP to input angles of triangle and check whether triangle is valid or not.

#include <stdio.h>

int main()

{

int angle1, angle2, angle3, sum;

printf("Enter three angles of triangle: \n");

scanf("%d%d%d", &angle1, &angle2, &angle3);

sum = angle1 + angle2 + angle3;

if(sum == 180 && angle1 > 0 && angle2 > 0 && angle3 > 0)

{

printf("Triangle is valid.");

}

else

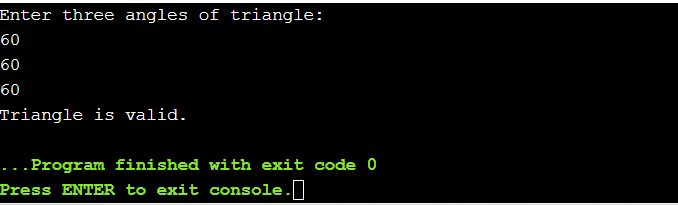
{

printf("Triangle is not valid.");

}

return 0;

}



17) : WAP to input all sides of triangle and check whether is triangle or not.

#include <stdio.h>

int main()

{

int side1, side2, side3;

printf("Enter three sides of triangle: \n");

scanf("%d%d%d", &side1, &side2, &side3);

if((side1 + side2) > side3)

{

if((side2 + side3) > side1)

{

if((side1 + side3) > side2)

{

printf("Triangle is valid.");

}

else

{

printf("Triangle is not valid.");

}

}

else

{

printf("Triangle is not valid.");

}

}

else

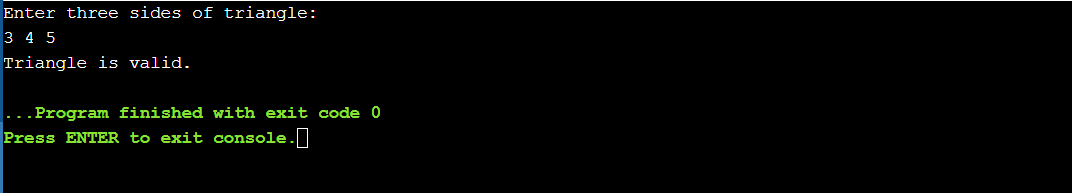
{

printf("Triangle is not valid.");

}

return 0;

}



18): WAP to check whether a triangle is equilateral, isosceles or scalene.

#include <stdio.h>

int main()

{

int sidea, sideb, sidec;

printf("enter three sides of triangle:");

scanf("%d %d %d" , &sidea, &sideb, &sidec );

if(sidea==sideb && sideb==sidec)

{

printf("This is an equilateral triangle.\n");

}

else if(sidea==sideb || sidea==sidec || sideb==sidec)

{

printf("This is an isosceles triangle.\n");

}

else

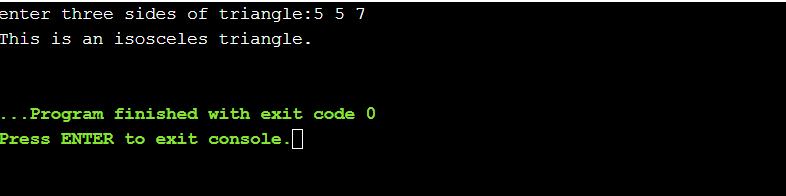
{

printf("This is a scalene triangle.\n");

}

return 0;

}



19): WAP to find all roots of quadratic equation.

#include <stdio.h>

#include <math.h>

int main()

{

int a, b, c, d;

double root1, root2;

printf("Enter a, b and c where a\*x\*x + b\*x + c = 0\n");

scanf("%d%d%d", &a, &b, &c);

d = b\*b - 4\*a\*c;

if (d < 0)

{

printf("First root = %.2lf + i%.2lf\n", -b/(double)(2\*a), sqrt(-d)/(2\*a));

printf("Second root = %.2lf - i%.2lf\n", -b/(double)(2\*a), sqrt(-d)/(2\*a));

}

else

{

root1 = (-b + sqrt(d))/(2\*a);

root2 = (-b - sqrt(d))/(2\*a);

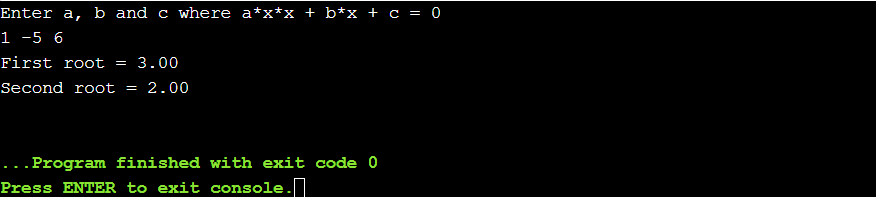
printf("First root = %.2lf\n", root1);

printf("Second root = %.2lf\n", root2);

}

return 0;

}



19) WAP to find all the roots of a quadratic equation.

#include <math.h>

#include <stdio.h>

int main() {

double a, b, c, discriminant, root1, root2, realPart, imagPart;

printf("Enter coefficients a, b and c: ");

scanf("%lf %lf %lf", &a, &b, &c);

discriminant = b \* b - 4 \* a \* c;

if (discriminant > 0) {

root1 = (-b + sqrt(discriminant)) / (2 \* a);

root2 = (-b - sqrt(discriminant)) / (2 \* a);

printf("root1 = %.2lf and root2 = %.2lf", root1, root2);

}

else if (discriminant == 0) {

root1 = root2 = -b / (2 \* a);

printf("root1 = root2 = %.2lf;", root1);

}

else {

realPart = -b / (2 \* a);

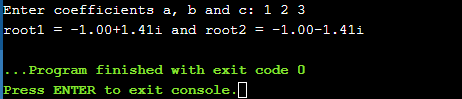
imagPart = sqrt(-discriminant) / (2 \* a);

printf("root1 = %.2lf+%.2lfi and root2 = %.2f-%.2fi", realPart, imagPart, realPart, imagPart);

}

return 0;

}



20) WAP to calculate profit and loss.

#include <stdio.h>

int main()

{

int cp,sp, amt;

printf("Enter cost price: ");

scanf("%d", &cp);

printf("Enter selling price: ");

scanf("%d", &sp);

if(sp > cp)

{

/\* Calculate Profit \*/

amt = sp - cp;

printf("Profit = %d", amt);

}

else if(cp > sp)

{

/\* Calculate Loss \*/

amt = cp - sp;

printf("Loss = %d", amt);

}

else

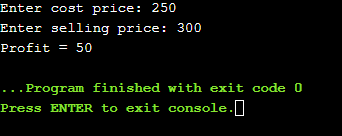
{

printf("No Profit No Loss.");

}

return 0;

}



21) WAP to input marks of 5 subjects Physics, Chemistry, Biology, Mathematics, Computers.

#include<stdio.h>

void main()

{

int marks;

printf("Enter your marks ");

scanf("%d",&marks);

if(marks<0 || marks>100)

{

printf("Wrong Entry");

}

else if(marks<40)

{

printf("Grade F");

}

else if(marks>=40 && marks<60)

{

printf("Grade E");

}

else if(marks>=60 && marks<70)

{

printf("Grade D");

}

else if(marks>=70 && marks<80)

{

printf("Grade C");

}

else if(marks>=80 && marks<90)

{

printf("Grade B");

}

else

{

printf("Grade A");

}

}

95

95

97

98

90

Enter five subjects marks: 95

95

95

96

97

Percentage = 95.60

Grade A

**...Program finished with exit code 0**

**Press ENTER to exit console.**

Percentage = 95.00

Grade A

22) WAP to calculate the gross salary of a person.

#include <stdio.h>

int main()

{

float basic, gross, da, hra;

printf("Enter basic salary of an employee: ");

scanf("%f", &basic);

if(basic <= 10000)

{

da = basic \* 0.8;

hra = basic \* 0.2;

}

else if(basic <= 20000)

{

da = basic \* 0.9;

hra = basic \* 0.25;

}

else

{

da = basic \* 0.95;

hra = basic \* 0.3;

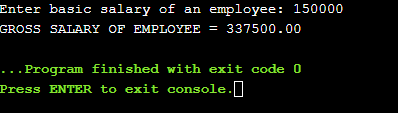
}

gross = basic + hra + da;

printf("GROSS SALARY OF EMPLOYEE = %.2f", gross);

return 0;

}



23) WAP to input electricity unit charges and calculate total electricity bill.

#include <stdio.h>

int main()

{

int unit;

float amt, total\_amt, sur\_charge;

printf("Enter total units consumed: ");

scanf("%d", &unit);

if(unit <= 50)

{

amt = unit \* 0.50;

}

else if(unit <= 150)

{

amt = 25 + ((unit-50) \* 0.75);

}

else if(unit <= 250)

{

amt = 100 + ((unit-150) \* 1.20);

}

else

{

amt = 220 + ((unit-250) \* 1.50);

}

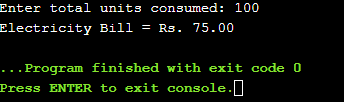
sur\_charge = amt \* 0.20;

total\_amt = amt + sur\_charge;

printf("Electricity Bill = Rs. %.2f", total\_amt);

return 0;

}



C Functions Exercises:

1. WAP to find cube of any number.

#include <stdio.h>

double cube(double num);

int main()

{

int num;

double c;

printf("Enter any number: ");

scanf("%d", &num);

c = cube(num);

printf("Cube of %d is %.2f", num, c);

return 0;

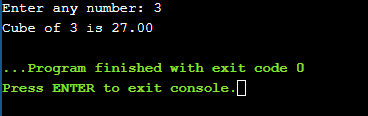
}

double cube(double num)

{

return (num \* num \* num);

}



2) WAP to find diameter, circumference and area of a circle.

#include <stdio.h>

#include <math.h>

double getDiameter(double radius);

double getCircumference(double radius);

double getArea(double radius);

int main()

{

float radius, dia, circ, area;

printf("Enter radius of circle: ");

scanf("%f", &radius);

dia = getDiameter(radius);

circ = getCircumference(radius);

area = getArea(radius);

printf("Diameter of the circle = %.2f units\n", dia);

printf("Circumference of the circle = %.2f units\n", circ);

printf("Area of the circle = %.2f sq. units", area);

return 0;

}

double getDiameter(double radius)

{

return (2 \* radius);

}

double getCircumference(double radius)

{

return (2 \* M\_PI \* radius); // M\_PI = PI = 3.14 ...

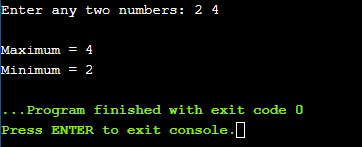
}

double getArea(double radius)

{

return (M\_PI \* radius \* radius); // M\_PI = PI = 3.14 ...

}



3) WAP to find maximum and minimum of two numbers.

#include <stdio.h>

int max(int num1, int num2);

int min(int num1, int num2);

int main()

{

int num1, num2, maximum, minimum;

printf("Enter any two numbers: ");

scanf("%d%d", &num1, &num2);

maximum = max(num1, num2);

minimum = min(num1, num2);

printf("\nMaximum = %d\n", maximum);

printf("Minimum = %d", minimum);

return 0;

}

int max(int num1, int num2)

{

return (num1 > num2 ) ? num1 : num2;

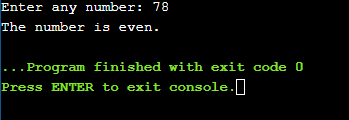
}

int min(int num1, int num2)

{

return (num1 > num2 ) ? num2 : num1;

}



4) WAP to check whether a number is even or odd.

#include <stdio.h>

int isEven(int num)

{

return !(num & 1);

}

int main()

{

int num;

printf("Enter any number: ");

scanf("%d", &num);

if(isEven(num))

{

printf("The number is even.");

}

else

{

printf("The number is odd.");

}

return 0;

}

Enter any number: 678

The number is even.

**...Program finished with exit code 0**

**Press ENTER to exit console.**

5) WAP to check whether a number is prime, Armstrong or perfect.

#include <stdio.h>

#include <math.h>

int isPrime(int num);

int isArmstrong(int num);

int isPerfect(int num);

int main()

{

int num;

printf("Enter any number: ");

scanf("%d", &num);

if(isPrime(num))

{

printf("%d is Prime number.\n", num);

}

else

{

printf("%d is not Prime number.\n", num);

}

if(isArmstrong(num))

{

printf("%d is Armstrong number.\n", num);

}

else

{

printf("%d is not Armstrong number.\n", num);

}

if(isPerfect(num))

{

printf("%d is Perfect number.\n", num);

}

else

{

printf("%d is not Perfect number.\n", num);

}

return 0;

}

int isPrime(int num)

{

int i;

for(i=2; i<=num/2; i++)

{

if(num%i == 0)

{

return 0;

}

}

return 1;

}

int isArmstrong(int num)

{

int lastDigit, sum, originalNum, digits;

sum = 0;

originalNum = num;

digits = (int) log10(num) + 1;

while(num > 0)

{

lastDigit = num % 10;

sum = sum + round(pow(lastDigit, digits));

num = num / 10;

}

return (originalNum == sum);

}

int isPerfect(int num)

{

int i, sum, n;

sum = 0;

n = num;

for(i=1; i<n; i++)

{

/\* If i is a divisor of num \*/

if(n%i == 0)

{

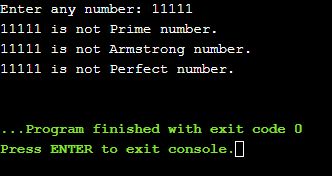
sum += i;

}

}

return (num == sum);

}



**Q.6 Write a C program to find all prime numbers between given interval using functions.**

#include <stdio.h>

int isPrime(int num);

void printPrimes(int lowerLimit, int upperLimit);

int main()

{

int lowerLimit, upperLimit;

printf("Enter the lower and upper limit to list primes: ");

scanf("%d%d", &lowerLimit, &upperLimit);

printPrimes(lowerLimit, upperLimit);

return 0;

}

void printPrimes(int lowerLimit, int upperLimit)

{

printf("All prime number between %d to %d are: ", lowerLimit, upperLimit);

while(lowerLimit <= upperLimit)

{

if(isPrime(lowerLimit))

{

printf("%d, ", lowerLimit);

}

lowerLimit++;

}

}

int isPrime(int num) {

int i;

for(i=2; i<=num/2; i++)

{

if(num % i == 0)

{

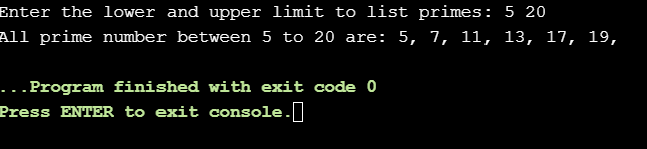
return 0;

}

}

return 1;

}



**Q.7 Write a C program to print all strong numbers between given interval using functions.**

#include <stdio.h>

long long fact(int num);

void printStrongNumbers(int start, int end);

int main()

{

int start, end;

printf("Enter the lower limit to find strong number: ");

scanf("%d", &start);

printf("Enter the upper limit to find strong number: ");

scanf("%d", &end);

printf("All strong numbers between %d to %d are: \n", start, end);

printStrongNumbers(start, end);

return 0;

}

void printStrongNumbers(int start, int end)

{

long long sum;

int num;

while(start != end)

{

sum = 0;

num = start;

while(num != 0)

{

sum += fact(num % 10);

num /= 10;

}

if(start == sum)

{

printf("%d, ", start);

}

start++;

}

}

long long fact(int num)

{

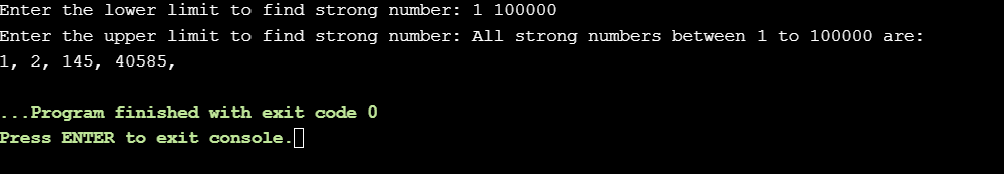
if(num == 0)

return 1;

else

return (num \* fact(num-1));

}

****

**Q.8 Write a C program to print all Armstrong numbers between given interval using functions.**

#include <stdio.h>

int isArmstrong(int num);

void printArmstrong(int start, int end);

int main()

{

int start, end;

printf("Enter lower limit to print armstrong numbers: ");

scanf("%d", &start);

printf("Enter upper limit to print armstrong numbers: ");

scanf("%d", &end);

printf("All armstrong numbers between %d to %d are: \n", start, end);

printArmstrong(start, end);

return 0;

}

int isArmstrong(int num)

{

int temp, lastDigit, sum;

temp = num;

sum = 0;

while(temp != 0)

{

lastDigit = temp % 10;

sum += lastDigit \* lastDigit \* lastDigit;

temp /= 10;

}

if(num == sum)

return 1;

else

return 0;

}

void printArmstrong(int start, int end)

{

while(start <= end)

{

if(isArmstrong(start))

{

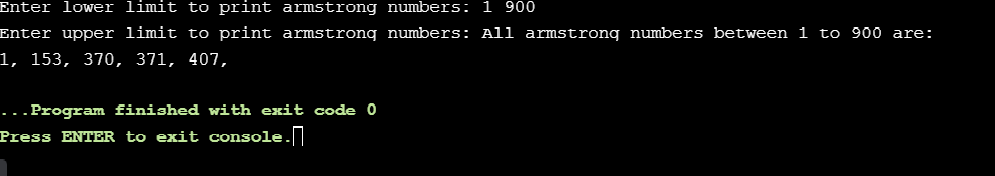
printf("%d, ", start);

}

start++;

}

}



**Q.9 Write a C program to print all perfect numbers between given interval using functions.**

#include <stdio.h>

int isPerfect(int num);

void printPerfect(int start, int end);

int main()

{

int start, end;

printf("Enter lower limit to print perfect numbers: ");

scanf("%d", &start);

printf("Enter upper limit to print perfect numbers: ");

scanf("%d", &end);

printf("All perfect numbers between %d to %d are: \n", start, end);

printPerfect(start, end);

return 0;

}

int isPerfect(int num)

{

int i, sum;

sum = 0;

for(i=1; i<num; i++)

{

if(num % i == 0)

{

sum += i;

}

}

if(sum == num)

return 1;

else

return 0;

}

void printPerfect(int start, int end)

{

while(start <= end)

{

if(isPerfect(start))

{

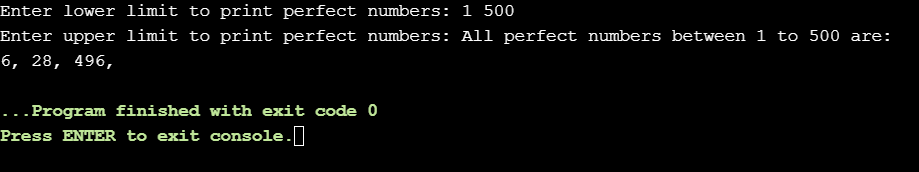
printf("%d, ", start);

}

start++;

}

}



**Q.10 Write a C program to find power of any number using recursion.**

#include <stdio.h>

double pow(double base, int expo);

int main()

{

double base, power;

int expo;

printf("Enter base: ");

scanf("%lf", &base);

printf("Enter exponent: ");

scanf("%d", &expo);

power = pow(base, expo);

printf("%.2lf ^ %d = %f", base, expo, power);

return 0;

}

double pow(double base, int expo)

{

/\* Base condition \*/

if(expo == 0)

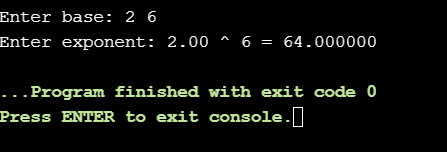
return 1;

else if(expo > 0)

return base \* pow(base, expo - 1);

else

return 1 / pow(base, -expo); }



**Q.11 Write a C program to print all natural numbers from 1 to n using recursion.**

#include <stdio.h>

void printNaturalNumbers(int lowerLimit, int upperLimit);

int main()

{

int lowerLimit, upperLimit;

printf("Enter lower limit: ");

scanf("%d", &lowerLimit);

printf("Enter upper limit: ");

scanf("%d", &upperLimit);

printf("All natural numbers from %d to %d are: ", lowerLimit, upperLimit);

printNaturalNumbers(lowerLimit, upperLimit);

return 0;

}

void printNaturalNumbers(int lowerLimit, int upperLimit)

{

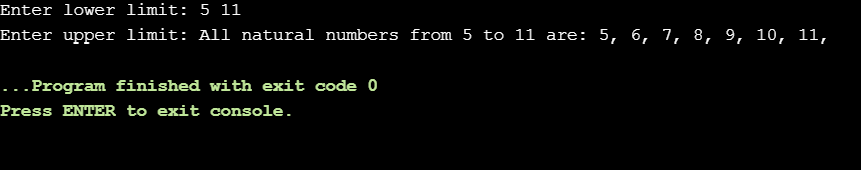
if(lowerLimit > upperLimit)

return;

printf("%d, ", lowerLimit);

printNaturalNumbers(lowerLimit + 1, upperLimit);

}



**Q.12 Write a C program to print all odd or even numbers in a given range using recursion.**

#include <stdio.h>

void printEvenOdd(int cur, int limit);

int main()

{

int lowerLimit, upperLimit;

printf("Enter lower limit: ");

scanf("%d", &lowerLimit);

printf("Enter upper limit: ");

scanf("%d", &upperLimit);

printf("Even/odd Numbers from %d to %d are: ", lowerLimit, upperLimit);

printEvenOdd(lowerLimit, upperLimit);

return 0;

}

void printEvenOdd(int cur, int limit)

{

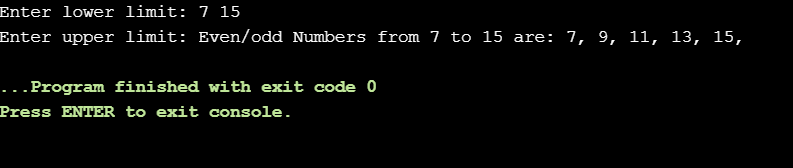
if(cur > limit)

return;

printf("%d, ", cur);

printEvenOdd(cur + 2, limit);

}



**Q.13 Write a C program to find sum of all natural numbers between 1 to n using recursion.**

#include <stdio.h>

int sumOfNaturalNumbers(int start, int end);

int main()

{

int start, end, sum;

printf("Enter lower limit: ");

scanf("%d", &start);

printf("Enter upper limit: ");

scanf("%d", &end);

sum = sumOfNaturalNumbers(start, end);

printf("Sum of natural numbers from %d to %d = %d", start, end, sum);

return 0;

}

int sumOfNaturalNumbers(int start, int end)

{

if(start == end)

return start;

else

return start + sumOfNaturalNumbers(start + 1, end);

}

Enter upper limit: 3

Sum of natural numbers from 2 to 3 = 5

**...Program finished with exit code 0**

**Press ENTER to exit console.**

14. Find the sum of even or odd in given interval using recursion:

#include <stdio.h>

int sumOfEvenOdd(int start, int end);

int main()

{

int start, end, sum;

printf("Enter lower limit: ");

scanf("%d", &start);

printf("Enter upper limit: ");

scanf("%d", &end);

printf("Sum of even/odd numbers between %d to %d = %d\n", start, end, sumOfEvenOdd(start, end));

return 0;

}

int sumOfEvenOdd(int start, int end)

{

if(start > end)

return 0;

else

return (start + sumOfEvenOdd(start + 2, end));

}

OUTPUT:

Enter lower limit: 1

Enter upper limit: 100

Sum of even/odd numbers between 1 to 100 = 2500

15. Find the reverse of number using recursion:

#include <stdio.h>

int main(){

int num,reverse\_number;

printf("\nEnter any number:");

scanf("%d",&num);

reverse\_number=reverse\_function(num);

printf("\nAfter reverse the no is :%d",reverse\_number);

return 0;

}

int sum=0,rem;

reverse\_function(int num){

if(num){

rem=num%10;

sum=sum\*10+rem;

reverse\_function(num/10);

}

else

return sum;

return sum;

}

OUTPUT:  
Enter any number:259

After reverse the no is :952

16. Find weather the number is palindrome or not

#include <stdio.h>

#include <math.h>

int reverse(int num);

int isPalindrome(int num);

int main()

{

int num;

printf("Enter any number: ");

scanf("%d", &num);

if(isPalindrome(num) == 1)

{

printf("%d is palindrome number.\n", num);

}

else

{

printf("%d is NOT palindrome number.\n", num);

}

return 0;

}

int isPalindrome(int num)

{

if(num == reverse(num))

{

return 1;

}

return 0;

}

int reverse(int num)

{

int digit = (int)log10(num);

if(num == 0)

return 0;

return ((num%10 \* pow(10, digit)) + reverse(num/10));

}

OUTPUT:

Enter any number: 1221

1221 is palindrome number.

17. Find the sum of digits of given number:

#include <stdio.h>

int sum (int a);

int main()

{

int num, result;

printf("Enter the number: ");

scanf("%d", &num);

result = sum(num);

printf("Sum of digits in %d is %d\n", num, result);

return 0;

}

int sum (int num)

{

if (num != 0)

{

return (num % 10 + sum (num / 10));

}

else

{

return 0;

}

}

OUTPUT:

Enter the number: 25                                                      Sum of digits in 25 is 7

18. Find the factorial of a number using recursion:

#include <stdio.h>

long int multiplyNumbers(int n);

int main() {

int n;

printf("Enter a positive integer: ");

scanf("%d",&n);

printf("Factorial of %d = %ld", n, multiplyNumbers(n));

return 0;

}

long int multiplyNumbers(int n) {

if (n>=1)

return n\*multiplyNumbers(n-1);

else

return 1;

}

OUTPUT:

Enter a positive integer: 5

Factorial of 5 = 120

19. Find the fibonaci serious using recursion:

#include <stdio.h>

void printFibonacci(int n){

static int n1=0,n2=1,n3;

if(n>0){

n3 = n1 + n2;

n1 = n2;

n2 = n3;

printf("%d ",n3);

printFibonacci(n-1);

}

}

int main(){

int n;

printf("Enter the number of elements: ");

scanf("%d",&n);

printf("Fibonacci Series: ");

printf("%d %d ",0,1);

printFibonacci(n-2);

return 0;

}

OUTPUT:

Enter the number of elements: 3

Fibonacci Series: 0 1 1

20. FIND the HCF of 2 numbers using recursion:

#include <stdio.h>

int hcf(int n1, int n2);

int main() {

int n1, n2;

printf("Enter two positive integers: ");

scanf("%d %d", &n1, &n2);

printf("G.C.D of %d and %d is %d.", n1, n2, hcf(n1, n2));

return 0;

}

int hcf(int n1, int n2) {

if (n2 != 0)

return hcf(n2, n1 % n2);

else

return n1;

}

OUTPUT:

Enter two positive integers: 6 24

G.C.D of 6 and 24 is 6.

21. Find the LCM of a given numbers using recursion:

#include <stdio.h>

int lcm(int a, int b);

int main()

{

int num1, num2, LCM;

printf("Enter any two numbers to find lcm: ");

scanf("%d%d", &num1, &num2);

if(num1 > num2)

LCM = lcm(num2, num1);

else

LCM = lcm(num1, num2);

printf("LCM of %d and %d = %d", num1, num2, LCM);

return 0;

}

int lcm(int a, int b)

{

static int multiple = 0;

multiple += b;

if((multiple % a == 0) && (multiple % b == 0))

{

return multiple;

}

else

{

return lcm(a, b);

}

}  
OUTPUT:

Enter two positive integers: 6 24

G.C.D of 6 and 24 is 6