Type 1. List of Java Programs: For Beginners

1. WAP to swap two numbers with using and without using third variable.

2. Write a menu driven program to check whether a triangle is valid, equilateral, isosceles, scalene or right angled or not.

3. Write a program to find the largest and smallest numbers among three numbers.

4. WAP to check whether an input character is vowel or not.

5. WAP to print factorial and table of a number.

6. WAP to find sum of the following series

a. 12 + 22 +32 +42 +52 +62 +72 +…. +n
b. 1-2+3-4+5-6+…… n
c. 1! + 2! + 3! + 4! +…… n
d. 2/9-5/13+8/17.

7. WAP a program to print table and factorial of the numbers lying between 1 to 1000.

8. Write a menu driven program to check whether a number is Prime, Armstrong, Magic, Palindrome, or Perfect or Not.

9. Write a menu driven program to count no of even digits, no of odd digits present in the number.

10. Write a menu driven program to print Fibonacci and Tribonacci series up to n terms.

11. Write a menu driven program to find even and odd divisors of a number.

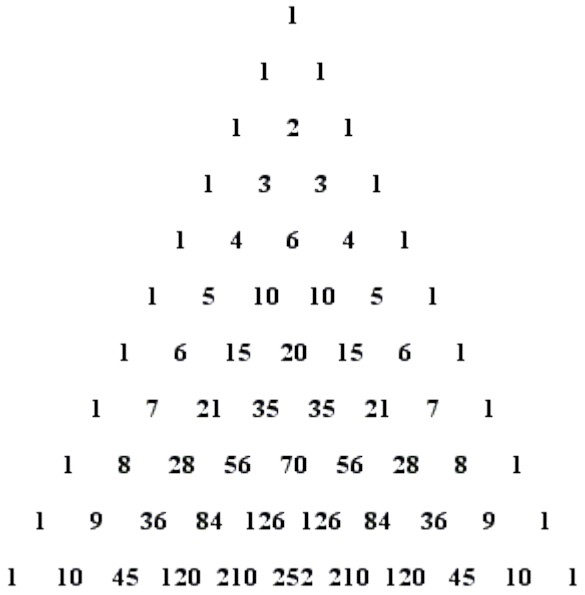
12. WAP to find roots of the quadratic equations.

13. WAP to check whether three points lies in the same line or not.

14. Write a menu driven program to find LCM and GCD of two numbers.

15. WAP to find the largest integer value n, such that its factorial can be correctly stored in the variable long.

16. WAP to generate pascal triangle.



17. WAP to print prime numbers from 1 to 1000.

18. WAP to check whether a point lies in the given circle or not.

19. WAP to decide whether a given date is valid date for a non leap year. Read only date and month.

20. WAP to check whether an input number is smith or not?

Where smith number is a composite number, the sum of whose digits is the sum of the prime factors obtained as a result of prime factorization (excluding 1). The first few numbers are 4,22, 27,58…
Example: a. 666 Prime factors are 2,3,3 and 37 Sum of the digits are (6+6+6) = 18. Sum of the digits of the factors (2 + 3 + 3 + (3 + 7)) = 18. So 666 is the Smith Number.

21. WAP to input a number and print it in words For ex. Input is 45 Output will be Four Five.

22. WAP to find Primorial of a number.
Primorial is defined as the product of prime numbers less than or equal to that number. e.g. Primorial of 6 will be 2*3*5 = 30.

23. WAP to print prime factors of a number.

24. WAP to print nth term of the Fibonacci series and also print the prime numbers present in the Fibonacci series.

25. WAP to print factorial of every number that comes in the every step while printing the table of any number.

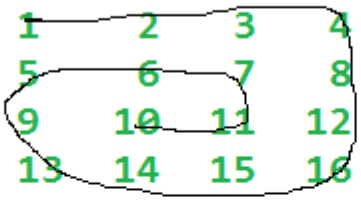
26. WAP to find sum of the prime numbers present in the single dimensional array of size n.

27. WAP to Check if a Matrix is Invertible.

28. WAP to compute determinant of a matrix.

29. WAP to print a Matrix in to spiral order.

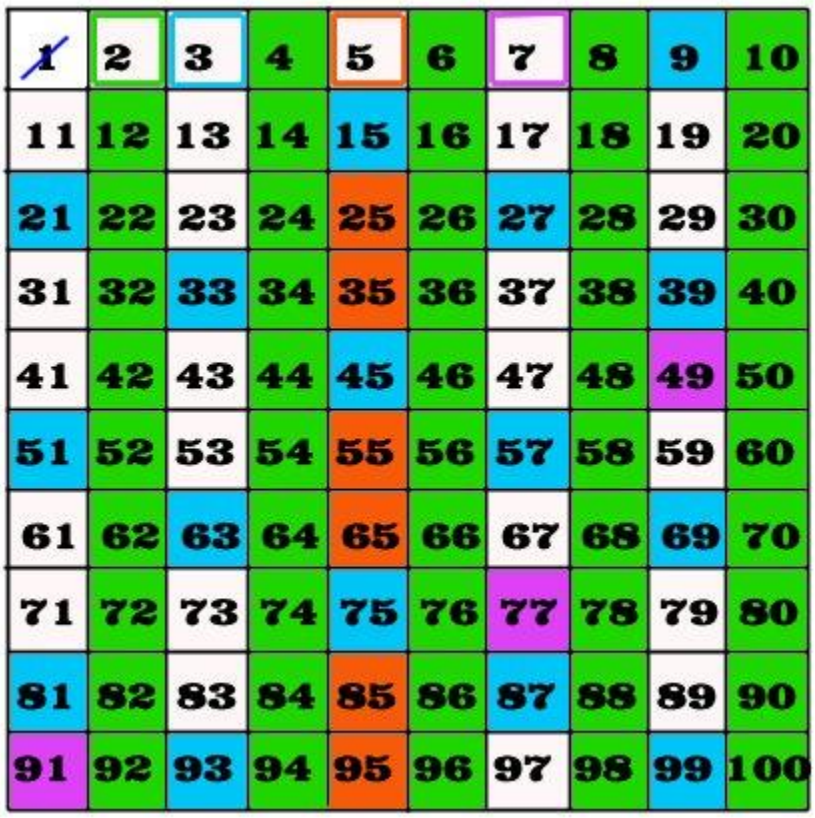
Input:



Output:

```
1 2 3 4 8 12 16 15 14 13 9 5 6 7 11 10
```

30. WAP to implement sieve of erathostone algorithm to generate prime numbers.



31. WAP to find transpose of a matrix.

32. WAP to print primary and secondry diagonal of a square matrix.

33. WAP to multiple two matrix of size m x n and p x q.

34. WAP that reads in an odd integer N and prints out N X N magic square. The square contains each of the integers between 1 and N^2 exactly once, such that all row sums, column sums and diagonal sums are equal.

35. WAP to generate Hadamard Matrix.

36. WAP to find sum of the digits of the largest and smallest number present in the array of size m x n.

37. WAP to convert single dimensional array to multi dimensional array and vice versa. As per users choice.

38. WAP a program to merge two single dimensional array of size m and p respectively.

39. Write a program to declare a square matrix A[][] of order (M x M) where M must be greater than 3 and less than 10. Allow the user to input positive integers into this matrix. Perform the following tasks on the matrix: (a) Sort the non-boundary elements in ascending order using any standard sorting technique and rearrange them in the matrix. (b) Calculate the sum of both the diagonals. (c) Display the original matrix, rearranged matrix and only the diagonal elements of the rearranged matrix with their sum.

40. Write a Program in Java to input a word and print its anagrams.

Note: Anagrams are words made up of all the characters present in the original word by re-arranging the characters. Example: Anagrams of the word TOP are: TOP, TPO, OPT, OTP, PTO and POT.

41. Write a program to input a word from the user and remove the duplicate characters present in it.

42. Write a Program in Java to fill a 2-D array with the first m x n prime numbers, where m is the number of rows and n is the number of columns.

43. Write a program to accept a sentence which may be terminated by either "." or "?" only. The words are to be separated by a single blank space. Print an error message if the input does not terminate with "." or "?". You can assume that no word in the sentence exceeds 15 characters, so that you get a proper formatted output.

Perform the following tasks: (i) Convert the first letter of each word to uppercase. (ii) Find the number of vowels and consonants in each word and display them with proper headings along with the words. 19.

44. Write a program to declare a square matrix A[ ][ ] of order MxM where M is the number of rows and the number of columns, such that M must be greater than 2 and less than 10. Accept the value of M as user input. Display an appropriate message for an invalid input. Allow the user to input integers into this matrix. Perform the following tasks: (a) Display the original matrix. (b) Rotate the matrix 90° clockwise as shown below: (c) Find the sum of the elements of the four corners of the matrix.

45. A simple encryption system uses a shifting process to hide a message. The value of the shift can be in the range 1 to 26. For example a shift of 7 means that A = U, B =V,C = W, etc. i e. Text : A B C D E F G H I J K L M N O P Q R S T U V W X Y Z Code: U V W X Y Z A B C D E F G H I J K L M N O P Q R S T Fist an extra space is added to the end of the string. To make things little more difficult, spaces within the original text are replaced with QQ before the text is encrypted. Double Q (QQ) was selected because no English word ends in Q or contains QQ. Additionally the coded message is printed in blocks of six characters

separated by spaces. The last block might not contain six characters. Write a program that takes the coded text (less than 100 characters), the shift value and prints the decoded original text. Your program must reject any non-valid value for shift and display an error message “INVALID SHIFT VALUE”. Assume all characters are upper case.
Test your program for the following data and some data that you have coded, using the rules given above:
SAMPLE DATA: 1. INPUT: CODED TEXT: “UHINBY LKKQCH HYLKK” SHIFT: 7 OUTPUT: DECODED TEXT: ANOTHER VALUE 2. INPUT: CODED TEXT: “RUIJGG EVGGBK SAGG” SHIFT: 11 OUTPUT: DECODED TEST: BEST OF LUCK 3. INPUT: CODED TEXT: “DKSMMW NAMMUK QMM” SHIFT: 29 OUTPUT: INVALID SHIFT VAULE.

46. WAP to count vowels, Upper Case Characters and Lower case characters present in the string.

47. WAP to count palindrome words present in the string.

48. WAP to count words that start and ends with vowel.

49. WAP to sort words present in the string lexicographically.

50. WAP to check whether any string contains digits or not.

51. WAP to count vowels present in each word of the string.

52. WAP to find the permutation of a string.

53. WAP a program to find longest palindrome in a string?

54. WAP to return highest occurred character in a String?

55. WAP to check if a String is valid shuffle of two String? Suppose You are given 3 strings: first, second, and third. third String is said to be a shuffle of first and second if it can be formed by interleaving the characters of first and second String in a way that maintains the left to right ordering of the characters from each string. For example, given first = "abc" and second = "def", third = "dabecf" is a valid shuffle since it preserves the character ordering of the two strings. So, given these 3 strings write a function that detects whether third String is a valid shuffle of first and second String.

56. Accept a paragraph of text consisting of sentences that are terminated by either ‘.’ (full stop), ‘!’ (exclamation mark) or a ‘?’ (question mark). Assume that there can be maximum 10 sentences in a paragraph. Write a program to arrange the sentences in increasing order of their number of words.

57. Write a program to declare a square matrix A[][] of order (M x M) where 'M' must be greater than 3 and less than 10. Allow the user to input positive integers into this matrix. Perform the following tasks on the matrix:
(a) Sort the boundary elements in descending order using any standard sorting technique and rearrange them in the matrix.

(b) Calculate the sum of the boundary elements.

(c) Display the original matrix, rearranged matrix and sum of the boundary elements.

Test your program with the sample data and some random data:

Example 1

INPUT :M = 4
9 2 1 5
8 13 8 4
15 6 3 11
7 12 23 8

OUTPUT:

ORIGINAL MATRIX
9 2 1 5
8 13 8 4
15 6 3 11
7 12 23 8
REARRANGED MATRIX
23 15 12 11
1 13 8 9
2 6 3 8
4 5 7 8

The sum of boundary elements is = 105

58. Write a program to accept the year, month and the weekday name of the 1st day of that month and generate its calendar.

59. A square matrix is said to be a Magic Square, if the sum of each row, each column and each diagonal is same. Write a program to enter an integer number 'n'. Create a magic square of size 'n*n'. Finally, print the elements of the matrix as Magic Square.

Note: n <= 5
Sample Input: Enter the size of the matrix : 4
Sample Output: The Magic Matrix of size 4×4 is:

16	2	3	13
5	11	10	8
9	7	6	12
4	14	15	1

Sample Input: Enter the size of the matrix : 5
Sample Output: The Magic Matrix of size 5×5 is:

17	24	1	8	15
23	5	7	14	16
4	6	13	20	22
10	12	19	21	3
11	18	25	2	9

60. Write a Program in Java to input a 2-D array of size ‘m*n’ and print its boundary (border) elements.
For example:

INPUT

1	2	3	4	5
6	7	8	9	10
11	12	13	14	15
16	17	18	19	20

OUTPUT

1	2	3	4	5
6				10
11				15
16	17	18	19	20

61. A simple encryption system uses a shifting process to hide a message. The value of the shift can be in the range 1 to 26. For example a shift of 7 means that A = U, B =V,C = W, etc.i e.

Text : A B C D E F G H I J K L M N O P Q R S T U V W X Y Z
Code: U V W X Y Z A B C D E F G H I J K L M N O P Q R S T

Fist an extra space is added to the end of the string. To make things little more difficult, spaces within the original text are replaced with QQ before the text is encrypted. Double Q (QQ) was selected because no English word ends in Q or contains QQ.

Additionally the coded message is printed in blocks of six characters separated by spaces. The last block might not contain six characters. Write a program that takes the coded text (less than 100 characters), the shift value and prints the decoded original text.Your program must reject any non-valid value for shift and display an error message "INVALID SHIFT VALUE)". Assume all characters are upper case. Test your program for the following data and some data that you have coded, using the rules given above:

SAMPLE DATA:

1. INPUT:
CODED TEXT : "UHINBY LKKQCH HYLKK"
SHIFT : 7
OUTPUT:
DECODED TEXT : ANOTHER VALUE

2. INPUT:
CODED TEXT : "RUIJGG EVGGBK SAGG"
SHIFT : 11
OUTPUT:
DECODED TEST : BEST OF LUCK

3. INPUT:
CODED TEXT : "DKSMMW NAMMUK QMM"
SHIFT : 29
OUTPUT:
INVALID SHIFT VAULE

62. Write a Program in Java to input a number in Decimal number system and convert it into its equivalent number in the Binary number system.

Note: Binary Number system is a number system which can represent a number in any other number system in terms of 0 and 1 only. This number system consists of only two basic digits i.e. 0 and 1.

For Example: 25 in the Decimal number system can be represented as 11001 in the Binary number system.

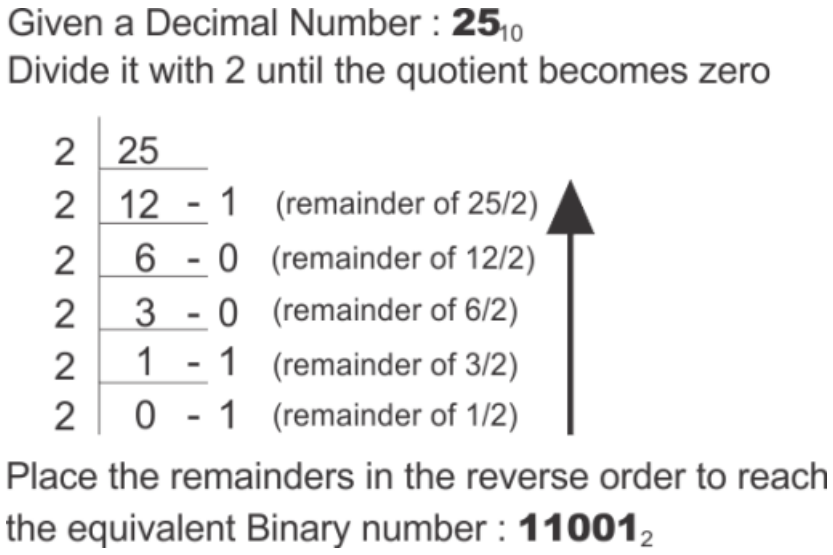


Figure Illustrating Decimal to Binary Number System Conversion

63. Write a Program in Java to print all the **Twin Prime** numbers within a given range.
Note: Twin Prime numbers are a pair of numbers which are both prime and their difference is 2.
Example:
Twin Prime numbers in the range 1 to 100 are :
(3,5) (5,7) (11,13) (17,19) (29,31) (41,43) (59,61) (71,73)

64. Caesar Cipher is an encryption technique which is implemented as ROT13 ('rotate by 13 places'). It is a simple letter substitution cipher that replaces a letter with the letter 13 places after it in the alphabets, with the other characters remaining unchanged.

ROT13

A/a	B/b	C/c	D/d	E/e	F/f	G/g	H/h	I/i	J/j	K/k	L/l	M/m
↕	↕	↕	↕	↕	↕	↕	↕	↕	↕	↕	↕	↕
N/n	O/o	P/p	Q/q	R/r	S/s	T/t	U/u	V/v	W/w	X/x	Y/y	Z/z

Write a program to accept a plain text of length L, where L must be greater than 3 and less than 100.
Encrypt the text if valid as per the Caesar Cipher.
Test your program with the sample data and some random data:
Example 1
INPUT : Hello! How are you?
OUTPUT : The cipher text is:
Uryyb? Ubj ner lbh?
Example 2
INPUT : Encryption helps to secure data.

OUTPUT : The cipher text is:
Rapelcgvba urycf gb frpher qngn.
Example 3
INPUT : You
OUTPUT : INVALID LENGTH

65. A **Circular Prime** is a prime number that remains prime under cyclic shifts of its digits. When the leftmost digit is removed and replaced at the end of the remaining string of digits, the generated number is still prime. The process is repeated until the original number is reached again.
A number is said to be prime if it has only two factors I and itself.

66. WAP to perform bubble sort and selection sort.

67. WAP to find common elements in two array.

68. WAP to reverse singly linked list.

69. WAP to find maximum repeated word in the file.

70.WAP to implement hash code and equals.

71. WAP to find longest substring without repeating characters.

72. WAP to sort a stack using temporary stack.

73. WAP to get distinct elements from an array by avoiding duplicate elements.

74. WAP for insertion sort in java.

75. WAP to check whether a given number is binary or not?

76. WAP to remove multiple spaces from the string.

77. WAP to remove HTML tag from the string.

78. WAP to find duplicate numbers from 1 to 1000.

79. WAP to implement linear and binary search.

80. WAP to change cases of the characters present in string.