COMPUTER SCIENCE CLASS XI: BATCH 2021

Practical file: This should contain 35 programs with proper documentation and output. [10 Marks]

LIST OF PROGRAMS FOR PRACRICAL WORK:

1. Write a program in java to store 20 numbers in a Single Dimensional Array and print the greatest and the smallest in the given set of numbers.
2. Write a program in java to store 10 numbers in a Single Dimensional Array. Arrange the numbers in descending order by using Bubble Sort Technique.
3. Write a program in java to store 10 numbers in a Single Dimensional Array. Arrange these numbers in ascending order by using Selection Sort Technique.
4. Write a program in java to store 10 numbers in a Single Dimensional Array. Now search, whether a given number is present or not in the array. If it is present then display 'Search successful' otherwise, display 'Number is not present'. Use Binary Search technique.
5. Write a program in java to store 10 numbers in a Single Dimensional Array. Now, shift each array element into the next cell in a cyclic order. Display the array elements after shifting them.
6. Write a program in java to store 10 numbers each, in two different Single Dimensional Arrays. Now, shift the elements in 'Z' form such that n[0] is shifted in m[0] and m[0] in n[1], n[1] in m[1] and so on …… . display the array elements after shifting them.
7. Write a program in java to store 10 numbers each, in two different Single Dimensional Array. Now search whether a number entered by the user is present or not. If present then display the number and also check whether it is prime number or not otherwise, display Number is not present. Use Linear Search technique.
8. Write a program in java to accept a decimal number (base 10) and convert it into its binary equivalent. Display the binary equivalent of the decimal number, entered by the user.
9. Write a program in java to store 10 numbers in a Single Dimensional Array. Display the numbers after eliminating duplicate numbers of the array.
10. Write a program in java to store 10 numbers each in two different Single Dimensional Arrays. Now, merge the numbers of both the arrays in a Single Dimensional Array. Display the elements of the merged array.
11. Write a program in java to store different numbers in 4 x 4 matrix in a Double Dimensional Array. Display the highest and the lowest numbers among the stored numbers in the matrix.

12. Write a program in java to store the numbers in 4 x 4 matrix in Double Dimensional Array. Display the sum of: (i) left diagonal elements (ii) right diagonal elements (iii) difference between the sum of left and right diagonal elements.
13. Transpose the matrix.
14. Replace each element of the left diagonal with zero.
15. Arrange the number of each row in ascending order and display the result.
16. Interchange the elements of first row with fourth row.
17. Display only the elements available above the left diagonal of the matrix.
18. Sum of border elements of the matrix.
19. Write a program to declare a matrix A[][] of order (M x N) where 'M' is the number of rows and 'N' is the number of columns such that the value of 'M' must be greater than 0 and less than 10 and the value of 'N' must be greater than 2 and less than 6. Allow the user to input digits (0 - 7) only at each location, such that each row represents an octal number.
20. Write a program to declare a single-dimensional array a[] and a square matrix b[][] of size N, where N > 2 and N < 10. Allow the user to input positive integers into the single dimensional array. Perform the following tasks on the matrix:

    1. Sort the elements of the single-dimensional array in ascending order using any standard sorting technique and display the sorted elements.

    2. Fill the square matrix b[][] in the following format: If the array a[] = {5, 2, 8, 1} then, after sorting a[] = {1, 2, 5, 8} Then, the matrix b[][] would fill as below:

1	2	5	8
1	2	5	1
1	2	1	2
1	1	2	5

    3. Display the filled matrix in the above format.

21. Write a program in java to accept a word and display the word in Piglatin Form. Aa word is said to be Piglatin by framing a new word with the first vowel present along with the remaining letters. The letters present before the first vowel are shifted towards the end followed by ‘ay’. Ex.- Input: trouble Output: oubletray.
22. Write a program to find the frequency of a word “the” present in a String. Input: Computer Science is the text book for the plus two students. Output: Frequency of ‘the’ = 2.
23. Write a program to find the penultimate word of a sentence. The penultimate word is the second last word present in a given sentence.
24. Write a program to accept any three-letter word and display all the probable three-letter combinations. No letter should be repeated in the output.
25. 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 and are in uppercase. Perform the following tasks:

    (a) Check for the validity of the accepted sentence.

    (b) Convert the non-palindrome words of the sentence into palindrome words by concatenating the word by its reverse (excluding the last character). Example: The reverse of the word HELP would be LEH (omitting the last alphabet) and by concatenating both, the new palindrome word is HELPLEH. Thus, the word HELP becomes HELPLEH. Note: The words which end with repeated alphabets, for example ABB would become ABBA and not ABBBA and XAZZZ becomes XAZZZAX. [Palindrome word: Spells same from either side. Example: DAD, MADAM etc.]

(c) Display the original sentence along with the converted sentence. Test your program for the following data and some random data:

Example - INPUT: THE BIRD IS FLYING.

OUTPUT: THE BIRD IS FLYING.

THEHT BIRDRIB ISI FLYINGNIYLF

26. Design a class Convert to find the date and the month from a given day number for a particular year. Example: If day number is 64 and the year is 2020, then the corresponding date would be: March 4, 2020 i.e. (31 + 29 + 4 = 64) Some of the members of the class are given below:

Classname : Convert

Data members/instance variables:

n :	integer to store the day number
d :	integer to store the day of the month (date)
m	integer to store the month
y :	integer to store the year

Methods/Member functions:

Convert ( ) : constructor to initialize the data members with legal initial values

void accept( ) : to accept the day number and the year

void day_to_date( ) : converts the day number to its corresponding date for a particular year and stores the date in ‘d’ and the month in ‘m’

void display( ) : displays the month name, date and year

Specify the class Convert giving details of the constructor( ),void accept( ), void day_to_date( ) and voiddisplay( ). Define a main( ) function to create an object and call the functions accordingly to enable the task.

27. A class Mix has been defined to mix two words, character by character, in the following manner: The first character of the first word is followed by the first character of the second word and so on. If the words are of different length, the remaining characters of the longer word are put at the end. Example: If the First word is “JUMP” and the second word is “STROLL”, then the required word will be “JSUTMRPOLL” Some of the members of the class are given below:

Classname : Mix

Data member/instance variable:

wrd : to store a word

len : to store the length of the word

Member functions/methods: Mix( ) : default constructor to initialize the data members with legal initial values

void feedword( ) : to accept the word in UPPER case void

mix_word(Mix P, Mix Q) : mixes the words of objects P and Q as stated above and stores the resultant word in the current object

void display( ) : displays the word

28. Design a class ArmNum to check if a given number is an Armstrong number or not. [A number is said to be Armstrong if sum of its digits raised to the power of length of the number is equal to the number] Example : 371 = 33 + 73 + 13 1634 = 14 + 64 + 34 + 44 54748 = 55 + 45 + 75 + 45 + 85 Thus 371, 1634 and 54748 are all examples of Armstrong numbers. Some of the members of the class are given below:

Class name : ArmNum
Data members/instance variables:
n : to store the number: to store the length of the number

Methods/Member functions:
ArmNum (int nn) : parameterized constructor to initialize the data member n=nn
int sum_pow(int i) : returns the sum of each digit raised to the power of the length of the number using recursive technique eg. 34 will return 32 + 42 (as the length of the number is 2)
void isArmstrong( ) : checks whether the given number is an Armstrong number by invoking the function sum_pow( ) and displays the result with an appropriate message

29. Design a class BinSearch to search for a particular value in an array. Some of the members of the class are given below:

Classname : BinSearch
Data members/instance variables:
arr[ ] : to store integer elements
n : integer to store the size of the array
Member functions/methods:
BinSearch(int nn) : parameterized constructor to initialize n=nn
void fillarray( ) : to enter elements in the array void sort( ) : sorts the array elements in ascending order using any standard sorting technique
int bin_search(int l,int u,int v) : searches for the value ‘v’ using binary search and recursive technique and returns its location if found otherwise returns -1
Define the class BinSearch giving details of the constructor( ), void fillarray( ), void sort( ) and int bin_search(int,int,int). Define the main( ) function to create an object and call the functions accordingly to enable the task.

Question 30

Design a program which takes two integer parameters namely number of the day (between 1 and 366) and the year (in 4 digits) as inputs and displays the date i.e. day, month and year.

Also find the corresponding date exactly after (N) days of the present date by accepting the value of (N) from the use, where the value of (N) is in the limit (1 <= N <= 100)

Design your program which will enable the output in the format given below:

Sample 1

```
INPUT:   DAY NUMBER          :   233
         YEAR                :   2020
         DATE  AFTER         :   17

OUTPUT:
         20TH.  AUGUST  2020
         DATE  AFTER 17 DAYS  :
         6TH.  SEPTEMBER  2020
```

Sample 2

```
INPUT:   DAY NUMBER          :   360
         YEAR                :   2020
         DATE  AFTER         :   45

OUTPUT:
         25TH.  DECEMBER  2020
         DATE  AFTER 45 DAYS  :
         8TH.  FEBRUARY  2021
```

Question 31

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 and are in lower case.

Perform the following tasks:

(a) Check for the validity of the accepted sentence and for the terminating character.

(b) Arrange the words contained in the sentence according to the size of the words in ascending order. If two words are of the same length then the first occurring comes first. The sentence should begin with a capital alphabet in both the cases i.e. Input and Output.

(c) Display both the sentences separately with each sentence beginning with a capital alphabet.

Design your program which will enable the output in the format given below:

Sample 1

INPUT: the lines are printed in reverse order.

OUTPUT:

```
The lines are printed in reverse order.
In the are lines order printed reverse.
```

Sample 2

INPUT: print the sentence in ascending order.

OUTPUT:

```
Print the sentence in ascending order.
In the print order sentence ascending.
```

Sample 3

INPUT: i love my country.

OUTPUT:

```
I love my country.
I my love country.
```

Question 32

A **MOBIUS** function M(N) returns the value -1 or 0 or 1 for a natural number (N) by the following conditions are defined :

When,

M (N) = 1 if N=1.

M (N) = 0 if any prime factor of N is contained more than once.

M (N) = $(-1)^{P}$ if N is the product of ‘P’ distinct prime factors.

Write a program to accept a positive natural number (N) and display the MOBIUS result with proper message.

Design your program which will enable the output in the format given below:

Sample 1

```
INPUT:      78
OUTPUT:     78  =  2  x  3  x  13
            NUMBER OF DISTINCT PRIME FACTORS = 3
            M(78)  =  -1
```

Sample 2

```
INPUT:      34
OUTPUT:     34  =  2  x  17
            NUMBER OF DISTINCT PRIME FACTORS  = 2
            M(34) = 1
```

Sample 3

```
INPUT:      12
OUTPUT:     12  =  2  x  2  x  3
            DUPLICATE  PRIME  FACTORS
            M(12) =  0
```

Sample 4

```
INPUT:      1
OUTPUT:     1  =  1
            NO PRIME FACTORS
            M(1) = 1
```