# NOPANY HIGH

**(I.C.S.E. & I.S.C Affiliated)**
**2-C, Nando Mullick Lane, Kolkata – 700006**

## Syllabus

**Academic Session: 2023 – 2024**

**Class: XII (SCIENCE) (Unit –I & II)**

| English Language | |
|---|---|
| **Text Book :**Total English for class XII<br>**Publisher: Morning Star Publishers** | |
| **Unit-I** | |
| **Test papers :** | Chapters 1 to 10.<br>All the topics according to ISC English Language question pattern. |
| | |
| | |
| | |
| **PROJECT** | **Will be discussed in class** |
| **Pre – Boards** | |
| **All chapters of Unit-I are included in the syllabus.** | |
| **Test Papers:** | Chapters 11 to 15.<br>All the topics according to ISC English Language question pattern. |
| | Last 5 years |

| English Literature | |
|---|---|
| **Text Book :** Echoes , Reverie , The Tempest<br>**Publisher: Evergreen Publishers** | |
| **Unit-I** | |
| **Echoes**<br>Chapter : 1 | Fritz by Satyajit Ray |
| Chapter : 2 | Quality by Joh Galaworthy |
| Chapter : 3 | The Story of an Hour by Kate Chopin |
| Chapter : 4<br>**Reverie**<br>Chapter : 1<br>Chapter : 2<br>Chapter : 6<br><br>**Drama :** | The Singing Lesson by Katherine Mansfield<br><br>The Darkling Thrush by Thomas Hardy ( Poetry )<br>Birches by Robert Frost ( Poetry)<br>John Brown by Bob Dylan ( Poetry )<br>The Tempest, Act 3 &4.<br>The Tepest (Act 4) |
| **PROJECT** | **Will be discussed in class** |
| **Pre – Boards** | |
| **All chapters of Unit-I are included in the syllabus.** | |

| | |
|---|---|
| **Echoes**<br>Chapter: 10 | <br>B.Wordsworth by V.S Naipaul. |
| **Reverie**<br>Chapter: 3<br>Chapter : 8 | <br>Dolphins by Carol Anne Duffy<br>Dover Beach by Matthew Arnold. |
| | |
| **Drama :** | The Tempest, Act 5. |

## Mathematics

### Unit-I

**Text Book:** ***ISC Mathematics (XII) (Volume I & II)***
***Publisher: M L Aggarwal***

**Section-A**

1.A. Relations and Functions:
B. Inverse Trigonometric Functions

2. Algebra
a. Determinant
b. Matrix

3.Calculus:
a. Continuity
b. Differentiation & Differentiability
c. Application of derivatives

4. Probability

**Section-B**

- Vectors
- 3 D Geometry

**Section-C**

- Linear Regression
- Linear Programming Problem

Project: As per council norms. Will be discussed in the class.

### Unit-II

**Text Book:** ***ISC Mathematics (XII) (Volume I & II)***
***Publisher: M L Aggarwal***
***All the chapters from UT-1 are included in UT-2***

**Section-A**

- Calculus:
  a. Integrals
  b. Differential Equation

**Section-B**

- Application of Integrals

**Section-C**

- Application of Calculus in Commerce and Economics

Project: As per council norms. Will be discussed in the class.

## Class: XII (Unit –I)

| Hindi Language ( हिन्दी व्याकरण ) | |
|---|---|
| **Text Book I** - *व्याकरण मंजूषा*<br>***Publisher:*** इंटर यूनिवर्सिटी प्रेस प्रकाशन | |
| निबंध -- | i. कंप्यूटर: वर्तमान समय की आवश्यकता<br>ii. समय का महत्व<br>iii. मन के हारे हार है, मन के जीते जीत ( मौलिक कहानी ) |
| अपठित गद्यांश -- | कोई दो ISC paper ke आधार पर |
| व्याकरण मंजूषा -- | मुहावरे<br>अशुद्ध वाक्य को शुद्ध करना |
| | |

| Hindi Literature ( हिन्दी साहित्य ) | |
|---|---|
| **Text Book I** : गद्य संकलन<br>**Text Book II** : काव्य मंजरी<br>***Publisher:*** *एवरग्रीन पब्लिकेशन*<br>***Text Book III:*** *सारा आकाश*<br>***Publisher:*** *इंटर यूनिवर्सिटी प्रेस प्रकाशन* | |
| काव्य मंजरी -- 1. तुलसीदास के पद<br>2. जाग तुझको दूर जाना<br>3. उद्यमी नर | |
| गद्य संकलन -- 1. क्या निराश हुआ जाए?<br>2. भक्तिन<br>3. संस्कृति क्या है? | |
| | |
| सारा आकाश -- उत्तरार्द्ध -- 1 से 7 तक | |

| Project- |
|---|
| महादेवी वर्मा का जीवन परिचय लिखते हुए ' भक्तिन ' कहानी में वर्णित समस्याओं पर प्रकाश डालते हुए एक |

| परियोजना कार्य बनाइए। |
|---|
| |

## Class: XII (Unit –II)

| Hindi Language ( हिन्दी व्याकरण ) | |
|---|---|
| निबंध – | i. वह दुर्घटना जिसे मैं भुला न पाई<br>ii. मेरा प्रिय कवि<br>iii. वर्नों का महत्व |
| अपठित गद्यांश – | कोई दो ISC paper के आधार पर |
| व्याकरण मंजूषा – | मुहावरे<br>अशुद्ध वाक्य को शुद्ध करना |
| | |

| Hindi Literature ( हिन्दी साहित्य ) | |
|---|---|
| काव्य मंजरी – 1. बादल को घिरते देखा है<br>2. अंधेरे का दीपक | |
| गद्य संकलन – 1. मजबूरी | |
| सारा आकाश – उत्तरार्द्ध – 8 से 10 तक | |
| ISC paper के आधार पर All Revision Work. | |

| Project- |
|---|
| तुलसीदास, रामधारी सिंह दिनकर, हरिवंश राय बच्चन कवियों में से किसी एक कवि का जीवन परिचय लिखते हुए साहित्य में उनके योगदान पर प्रकाश डालते हुए एक परियोजना कार्य बनाइए। |

## PHYSICS

### Unit-I

**Text Book : *Nootan ISC Physics for Class-XII***
**Publisher : Nageen Prakashan**

**Text Book for Physics Practical: *ISC Practical Physics (Vol-I) for Class XII***

**Reference Book : *ISC Physics,Class XI, VOL-I and II***
**Publisher : Kalyani Publications**

| | **THEORY** |
|---|---|
| Unit : 1<br>Chapter : 1<br>Chapter : 2<br>Chapter : 3<br>Chapter : 4 | ELECTROSTATICS<br>Electric Charges and Fields<br>Gauss' Theorm<br>Electric Potential<br>Capacitors and Dielectrics |
| Unit : 2<br>Chapter : 5<br>Chapter : 6 | ATOM AND NUCLEI<br>Electric Resistance and Ohm's Law<br>DC Circuits and Measurements |

| | |
|---|---|
| Unit : 3 | MAGNETIC EFFECTS OF CURRENT AND MAGNETISM |
| Chapter : 7 | Moving Charges and Magnetic Field |
| Chapter : 8 | Torque on a Current-Loop : Moving-Coil Galvanoment |
| Chapter : 9 | Magnetic field and Earth's Magnetism |
| Chapter : 10 | Magnetic Classification of Substances |
| Unit : 4 | ELECTROMAGNETIC INDUCTION AND ALTERNATING CURRENTS |
| Chapter : 11 | Electromagnetic Induction |
| Chapter : 12 | Alternating Current |
| Unit : 4 | ELECTROMAGNETIC WAVES |
| Chapter : 13 | Electromagnetic Waves |
| Unit : 3 | OPTICS |
| Chapter : 14 | Reflection of Light : Spherical Mirrors |
| Chapter : 15 | Refraction of Light at a Plane Interface : Total Internal Reflection : Optical Fibre |
| Chapter : 16 | Refraction of Light at Spherical Surfaces : Lenses |
| Chapter : 17 | Refraction and Dispersion of Light through a Prism |
| Chapter : 18 | Optical Instruments |
| Chapter : 19 | Wave nature of Light : Huygens' Principle |
| Chapter : 20 | Interference of Light |
| Chapter : 21 | Diffraction of Light |
| Chapter : 22 | Polarisation of Light |
| PROJECT | Will be discussed in the class |
| PRACTICAL | As per ISC guidelines |

## Unit-II

**Text Book : *Nootan ISC Physics for Class-XII***
**Publisher : Nageen Prakashan**
**Text Book for Physics Practical: *ISC Practical Physics (Vol-I) for Class XII***
**Reference Book : *ISC Physics,Class XI, VOL-I and II***
**Publisher : Kalyani Publications**
**All chapters of Unit-I are included in the syllabus.**

| | THEORY |
|---|---|
| Unit : 7 | DUAL NATURE OF RADIATION AND MATTER |
| Chapter 23 | Photoelectric Effect |
| Chapter 24 | Matter waves |
| Chapter 25 | X- rays |
| Unit : 8 | ATOM AND NUCLEI |
| Chapter 26 | Atom, Origin of spectra : Bohr's Theory of Hydrogen atom |
| Chapter 27 | Nuclear Structure |
| Chapter 28 | Radioactivity |
| Chapter 29 | Mass-Energy Equivalence : nuclear Binding Energy |
| Chapter 30 | Nuclear Fission and Nuclear Fusion : Sources of Energy |
| Unit : 9 | ELECTRONIC DEVICES |
| Chapter 31 | Semiconductor Electronics |
| Chapter 32 | Junction Diodes |
| Chapter 33 | Junction Transistors |
| Chapter 34 | Logic gates |

| | |
|---|---|
| Unit : 10<br>Chapter 35 | COMMUNICATION SYSTEMS<br>Communication Systems |
| | |
| PROJECT | Will be discussed in the class |
| PRACTICAL | As per ISC guidelines |

| **CHEMISTRY** | |
|---|---|
| ***Text Book :ISC Chemistry***<br>***Publisher: Kalyani***<br>***Author: K.L.Chugh***<br>***Practical Book :Nova ISC Chemistry Lab Manual (Class 12)***<br>***Publisher : Nova Publications*** | |
| ***Unit – I*** | |
| **Section – A : Physical Chemistry** | |
| Chapter : 1 | Solid State |
| Chapter : 2 | Solutions |
| Chapter : 3 | Electrochemistry |
| Chapter : 4 | Chemical Kinetics |
| **Section – B : Inorganic Chemistry** | |
| Chapter : 6 | General Principles and Processes of Isolation of Elements |
| Chapter :7 | p-Block Elements |
| **Section – C : Organic Chemistry** | |
| Chapter : 10 | Haloalkanes and Haloarenes |
| Chapter : 11 | Alcohols , Phenols and Ethers |
| Chapter : 12 | Aldehydes , Ketones and Carboxylic Acids |
| Chapter : 13 | Organic Compounds Containing Nitrogen |
| **Practicals:**<br>1).Titrations (Volumetric Analysis)<br>Introduction to Volumetric Analysis , Redox Titrations<br>2). Measurement of Rate of Reaction<br>3). Identification of functional groups and organic compounds.<br>4). Characteristic Tests of Carbohydrates and Proteins<br>5).Qualitative Inorganic Analysis :<br>Identification of single salt containing one anion and one cation<br>Anions : $CO_3^{2-}$ , $NO_2^-$ , $S^{2-}$ , $SO_3^{2-}$ , $SO_4^{2-}$ , $NO_3^-$ , $CH_3COO^-$ , $Cl^-$ , $Br^-$ , $I^-$ , $C_2O_4^{2-}$ , $PO_4^{3-}$<br>Cations: $NH_4^+$ , $Pb^{2+}$ , $Cu^{2+}$ , $Al^{3+}$ , $Zn^{2+}$ , $Mn^{2+}$ , $Ni^{2+}$ , $Co^{2+}$ , $Ba^{2+}$ , $Sr^{2+}$ , $Ca^{2+}$ , $Mg^{2+}$ | |
| ***Pre Board Exam*** | |
| ***All chapters of Unit – I are included in Pre BoardExam*** | |
| **Section – A : Physical Chemistry** | |
| Chapter :5 | Surface Chemistry |
| **Section – B : Inorganic Chemistry** | |
| Chapter : 8 | d- and f-Block Elements |
| Chapter :9 | Coordination Compounds |
| **Section – C : Organic Chemistry** | |
| Chapter : 14 | Biomolecules |
| Chapter : 15 | Polymers |
| Chapter :16 | Chemistry in Everyday Life |

| BIOLOGY | |
|---|---|
| **Text Book : ISC Srijan Biology XII**<br>**Publisher : Srijan Publishers Private Limited**<br>**Author : Veer Bala Rastogi** | |
| **Unit-I** | |
| Chapter : 1 | Reproduction in Organisms |
| Chapter : 2 | Sexual Reproduction in Flowering plants |
| Chapter : 3 | Human Reproduction |
| Chapter : 4 | Reproductive Health |
| Chapter : 5 | Principles of Inheritence |
| Chapter : 6 | Molecular Basis of Inheritence |
| Chapter : 8 | Human Health and Diseases |
| Chapter : 10 | Microbes in Human Welfare |
| Chapter : 11 | Biotechnology- Principles and Processes |
| Chapter : 12 | Biotechnology and its Applications |
| Chapter : 13 | Organisms and Population |
| Chapter : 14 | Ecosystem |
| Chapter : 15 | Biodiversity and Conservation |
| **PROJECT** | **Any one topic from ISC Class XII syllabus** |
| **Practical:** | **As per ISC guidlines** |
| **Preboard Examination** | |
| **All chapters of Unit-I are included in the syllabus.** | |
| Chapter : 7 | Evolution |
| Chapter : 9 | Stategies for Enhancement in Food Production |
| Chapter : 16 | Environmental issues |
| **PROJECT** | **Project topics same as the UNIT 1** |
| **Practical:** | **As per ISC guidlines** |

| Computer | |
|---|---|
| **Text Book :** Understanding ISC Computer Science (Java with BlueJ) Class XII<br>**Publisher :** Arya Publishing Company<br>**Author Name :** Vijay Kumar Pandey and Dilip Kumar Dey | |
| **Unit-I** | |
| Chapter : 1 | Boolean Algebra. |
| Chapter : 2 | Computer Hardware. |
| Chapter : 3 | Implementation of Algorithms to Solve Problems |
| Chapter : 4 | Objects and classes |

| | |
|---|---|
| Chapter : 5 | Data Types and Variables |
| Chapter : 6 | Statements and Scope |
| Chapter : 7 | String Manipuations |
| Chapter : 8 | Arrays |
| Chapter : 9 | Methods |
| | |
| **Unit-II** | |
| **All chapters of Unit-I are included in the syllabus.** | |
| Chapter : 10 | Class as User Defined and Constructors |
| Chapter : 11 | Recusion |
| Chapter : 12 | Inheritance, Intefaces amd Ploymorphism |
| Chapter : 13 | Data Structures |
| Chapter : 14 | Computational Complexity and Big O Notation. |

| **ENVIRONMENTAL SCIENCE** | |
|---|---|
| **Text Book- ISC Environmental Science**<br>**Publisher- Goyal Brother Prakashan** | |
| Theory( Paper-I)-70Marks | |
| | |
| Chapter- 1 | Human Beings and nature |
| Chapter- 2 | Population Ecology |
| Chapter -3 | Conservation Ecology |
| Chapter – 4 | Monitoring Pollution |
| Chapter – 5 | Third world development |
| Chapter -6 | Sustainable Agriculture |
| **Project: As instructed in Scope** | |
| **Project – 30 Marks( Paper-II)**<br>Address a current environmental problem(preferably at local or regional scale)and should include problem identification and analysis, use of secondary data as well as some collection of | |

## Project– 2023-2024

**Unit I**

1. Input a number and check whether it is a “Bouncy Number” or not.
   (A Bouncy Number is a number which is neither in increasing order nor in decreasing order).
   Sample Input : 774361
   Sample Output: Bouncy number.
   Sample Input : 774321
   Sample Output: Not a bouncy number.
2. Input a number and check whether it is a “Evil Number” or not.
   ( An Evil Number is a positive whole number which has even no. of 1’s in its binary equivalent)
   Sample Input : 9
   Sample output: It’s binary equivalent is 1001

No. of 1's : 2
9 is an Evil Number.

3. Input a number and check whether the number is a "Fascinating Number" or not.
( A Fascinating number are some three digit numbers which exhibit a very interesting property.)
When a number is multiplied by 2 and 3, and both these products are concatenated with the original number, all digits from 1 to 9 are present in the number exactly once, regardless of the no. of zero's).
Sample Input : n = 192
Sample Output : 192 * 1 = 192
192 * 2 = 384
192 * 3 = 576
Concatinating the numbers : 192384576
192 is a Fascinating number.
Other Fascinaing numbers are: 219,273,327,1902,1920..etc

4. Declare a Square Matrix A[][] of order (m*m) where 'm' must be greater than 3 and less than 10. Allow the user to input positive integers into the matrix. Perform the following tasks on the matrix.

a) Sort non-boundary elements in ascending order using any standard technique and rearrange them in the matrix.
b) Calculate sum of both the diagonals.
c) Display the original matrix and the rearranged matrix and only the diagonal elements of the rearranged matrix along with their sum.

5. A class Rearrange has been defined to modify a word by bringing all the vowels in the word at the beginning followed by the consonants.
Example:
ORIGINAL becomes OIIARGNL
Some of the members of the class are given below:
Class name: Rearrange
Data Member/instance variable:
wrd: to store a word
newwrd: to store the rearranged word
Member functions/methods:
Rearrange(): default constructor
void readword(): to accept the word in UPPER case
vow freq_vow_con(): finds the frequency of vowels and consonants in the word and displays them with an appropriate message
void arrange(): rearranges the word by bringing the vowels at the beginning followed by consonants
void display(): displays the original word along with the rearranged word
Specify the class Rearrange, giving the details of the constructor(), void readword(), void freq _vow_con(), void arrange() and void display(). Define the main() function to create an object and call the functions accordingly to enable the task.

6. Design a class MatRev to reverse each element of a matrix.
Example:

| | | |
|---|---|---|
| 72 | 371 | 5 |
| 12 | 6 | 426 |
| 5 | 123 | 94 |

becomes

| | | |
|---|---|---|
| 27 | 173 | 5 |
| 21 | 6 | 624 |
| 5 | 321 | 49 |

Some of the members of the class are given below:
Class name: MatRev
Data members/instance variables:
arr[][] : to store integer elements
m: to store the number of rows
n: to store the number of columns
Member functions/methods:
MatRev(int mm, intnn): parameterized constructor to initialise the data members m = mm and n = nn
void fillarray(): to enter elements in the array
int reverse(int x): returns the reverse of the number x
void revMat(MatRev P): reverses each element of the array of the parameterized object and stores it in the array of the current object
void show(): displays the array elements in matrix form.

Define the class MatRev giving details of the constructor ( ), void fillarray (), int reverse(int), void revMat(MatRev) and void show(). Define the main () function to create objects and call the functions accordingly to enable the task.

7. A **Goldbach** number is a positive even integer that can be expressed as the sum of two odd primes.

Note: All even integer numbers greater than 4 are Goldbach numbers.
**Example:**
6 = 3 + 3
10 = 3 + 7
10 = 5 + 5
Hence, 6 has one odd prime pair 3 and 3. Similarly, 10 has two odd prime pairs, i.e. 3 and 7, 5 and 5.
Write a program to accept an even integer 'N' where N > 9 and N < 50. Find all the odd prime pairs whose sum is equal to the number 'N'.
Test your program with the following data and some random data:
**Example 1**
**INPUT:**
N = 14
**OUTPUT:**
PRIME PAIRS ARE:
3, 11
7, 7
**Example 2**
**INPUT:**
N = 30
**OUTPUT:**
PRIME PAIRS ARE:
7, 23
11, 19
13, 17
**Example 3**
**INPUT:**
N = 17
**OUTPUT:**

INVALID INPUT. NUMBER IS ODD.

**Example 4**

**INPUT:**

N = 126

**OUTPUT:**

INVALID INPUT. NUMBER OUT OF RANGE.

8. Permutation and Combination of two numbers 'n' and 'r' are calculated as

$^{n}P_{r}$ = ! n / !(n-r)

$^{n}C_{r}$ = ! n/ !(n-r)* ! r

where Permutation is denoted as $^{n}P_{r}$ and Combination is denoted as $^{n}C_{r}$ . The $^{n}P_{r}$ means permutation of 'n' and 'r' and $^{n}C_{r}$ means combination of 'n' and 'r' .

Write a program to calculate and display the number of permutation and combination of two numbers 'n' and 'r' by using the above formula.

Sample Input : Enter the value of n : 11
Enter the value of r : 10

Sample Output : $^{n}P_{r}$ = 39916800
$^{n}C_{r}$ = 11.

9. Write a program in Java to accept a string. Count and display the frequency of each character present in the string. The character with multiple frequencies should be displayed only once.

Sample Input : **golden jubilee**

Sample Output :

| Alphabet | g | o | l | d | e | n | j | u | b | i |
|---|---|---|---|---|---|---|---|---|---|---|
| Frequency | 1 | 1 | 2 | 1 | 3 | 1 | 1 | 1 | 1 | 1 |

10. Write a program in Java to accept a String and display all the words present in the string in PigLatin form.

Example :

Sample Input : THE CAPITAL OF INDIA IS NEW DELHI.

Sample Output : ETHAY APITALCAY OFAY INDIAAY ISAY EWNAY ELHIDAY.

11. A bank intends to design a program to display the denomination of an input amount, upto 5 digits.The available denomination with the bank are of rupees 1000, 500, 100, 50, 20, 10, 5, 2 and 1.

Design a program to accept the amount from the user and display the break-up in descending order of denomination. (i.e. preference should be given to the highest denomination available) along with the total number of notes.[Note: Only the denomination used should be displayed]. Also print the amount in words according to the digits.

Example:

INPUT:
14856
OUTPUT:
ONE FOUR EIGHT FIVE SIX
DENOMINATION :
1000 * 14 = 14000
500 * 1 = 500
100 * 3 = 300
50 * 1 = 50
5 * 1=5
1*1=1
TOTAL =14856
TOTAL NUMBER OF NOTES=21.

12. Write a program to accept a sentence and display the new sentence according to the length of the word.
Sample Input : I Love My Country.
Sample Output : Country Love My I

13. Write a program to create a Double Dimensional Array of order [m ] * [m] to store integer numbers. Now, pass the array to a method Boundary() to calculate and display the sum of the boundary elements of the array.
Sample Input : Enter no. of rows and cols : 4
Original Matrix:

| | | | |
|---|---|---|---|
| 11 | 14 | 15 | 9 |
| 12 | 20 | 10 | 8 |
| 6 | 18 | 16 | 4 |
| 5 | 22 | 20 | 3 |

Sample Output: Sum of boundary elements: 129.

14. Accept a string . a word to be deleted along with its position no. delete the word and display the new sentence.
Sample Input : Enter a Sentence : As You Sow, So So Shall You Reap.
Enter word to be deleted : So
Enter position no. : 13
Sample Output : : As You Sow, So Shall You Reap.

15. 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.

| | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 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?

## Unit II

1. Design a class called *Change* to convert a decimal number into its equivalent number in base 16 and vice versa.

For Eg. I) The decimal number 35 is 23 in Hexadecimal
ii) The decimal number 107 is 6B in Hexadecimal

Some members of the class *Change* is as follows:

| | |
|---|---|
| Class name | : *Change* |
| **Data members/Instance Variabes :** | |
| a[] | : An Integer array. |
| n | : Integer to be converted to base 16. |
| **Member Functions** | : |
| Change() | : constructor to initialize 0 to instance variables. |
| void input() | : To accept an integer to convert |
| void hexa(String s) | : To convert hexadecimal number back to decimal |
| void decihexa() | : To convert decimal integer'n' to hexadecimal form. |

2. Given a matrix of n x n size, the task is to find the saddle point of the matrix. A saddle point is an element of the matrix such that it is the minimum element in its row and the maximum in its column.

**Examples :**

**Input:** Mat[3][3] = { {1, 2, 3},
{4, 5, 6},
{7, 8, 9}}

**Output:** 7
7 is minimum in its row and maximum in its column.

**Input:** Mat[3][3] = {{1, 2, 3},
{4, 5, 6},
{10, 18, 4}}

**Output:** No saddle point

3. An digit number x is called Keith number if it appears in a special sequence (defined below) generated using its digits. The special sequence has first n terms as digits of x and other terms are recursively evaluated as sum of previous n terms.
The task is to find if a given number is Keith Number or not.

**Examples:**

Input : x = 197

Output : Yes

197 has 3 digits, so n = 3

The number is Keith because it appears in the special

sequence that has first three terms as 1, 9, 7 and

remaining terms evaluated using sum of previous 3 terms.

1, 9, 7, 17, 33, 57, 107, **197**, .....

Input : x = 12
Output : No
The number is not Keith because it doesn't appear in
the special sequence generated using its digits.
1, 2, 3, 5, 8, 13, 21, .....

Input : x = 14
Output : Yes
14 is a Keith number since it appears in the sequence,
1, 4, 5, 9, **14**, ...

4. Input a paragraph containing 'n' number of sentences where 1<= n <4 . The words are to be separated with a single blank space and are in uppercase . A sentence may be terminated either with a full stop '.' Or a question mark '?' only. Any other character may be ignored. Perform the following operations :
   a) Accept the number of sentences. If the number of sentences exceeds the limit, an appropriate error message should be displayed.
   b) Find the no. of words in the whole paragraph.
   c) Display the words in ascending order of their frequency. Words with same frequency may appear in any order.

   Example 1:
   Input : Enter number of sentences = 1
   Enter the sentences : TO BE OR NOT TO BE.
   Output : Total number of Words : 6

| Word | Frequency |
|---|---|
| OR | 1 |
| NOT | 1 |
| TO | 2 |
| BE | 2 |

   Example 2:
   Input : Enter number of sentences = 3
   Enter the sentences : THIS IS A STRING PROGRAM. IS THIS EASY? YES, IT IS.
   Output : Total number of Words : 11

| Word | Frequency |
|---|---|
| A | 1 |
| STRING | 1 |
| PROGRAM | 1 |
| EASY | 1 |
| YES | 1 |
| IT | 1 |
| THIS | 2 |

IS 3

Example 3:

Input : Enter number of sentences = 5

Output : Sentence limit is 4.

5. A Class DeciOct has been defined to convert a decimal number into its equivalent octal number. Some of the members of the class are given below:

Class name :DeciOct

Data Members/ instance variables :

n : stores the decimal number.

oct : Stores the octal number.

Member Functions/ Methods :

DeciOct() : Constructor to initialize data members n and oct with 0.

void getnum(int nn) : assigns nn to n

void deci_oct() : Calculates the octal equivalent of 'n' and stores it in 'oct' using the **Recursive technique**.

void show() : displays the decimal number n, calls the function deci_oct() and displays its octal equivalent.

Specify the class DeciOct , giving details of the constructor, void getnum(int), void deci_oct() and void show(). Also define a main() function to create an object and call the function accordingly to enable the task.

6. Write a program in Java to store the elements in two different double dimensional arrays (in matrix form) a and B each of order 4*4. Find the product of both the matrices and store the result in matrix C. display the elements of matrix C.

Note : Two matrices can be multiplied only if the number of columns of the first matrix is equal to the number of rows of the second matrix.

Sample Input : Enter elements of Matrix A :

| | | | |
|---|---|---|---|
| 3 | 2 | 1 | 2 |
| 6 | 4 | 5 | 0 |
| 7 | -1 | 0 | 2 |
| 4 | 3 | 1 | 1 |

Enter elements of Matrix B:

| | | | |
|---|---|---|---|
| -2 | -4 | -1 | 0 |
| 3 | 6 | -5 | 2 |
| 5 | 3 | 4 | 6 |
| 0 | -2 | 2 | 5 |

Sample Output: Matrix C

| | | | |
|---|---|---|---|
| 5 | -1 | -5 | 20 |
| 25 | 15 | -6 | 38 |
| -17 | -38 | 2 | -8 |
| 6 | 3 | -13 | 17 |

7. 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

8. Given a time in numbers, we can convert it into words.

Eg. 5:00 – Five O' clock

5:10 – Ten minutes past five

5:15 – Quarter past five

5:30 – Half past five

5:40 – Twenty minutes to six

5:45 – Quarter to six

5:47 – Thirteen minutes to six.

Write a program which first inputs two integers, the first between 1 to 12(both inclusive) and second between 0 to 59 (both inclusive) and prints out the time they represent, in words. Your program should follow the format of the above example.

Sample Input: Enter time : 3,0

Sample output : 3:00 Three O' clock

Sample Input: Enter time : 7,29

Sample output : 7:29 Twenty nine minutes past seven

Sample Input: Enter time : 6,34

Sample output : 6:34 Twenty six minutes to seven

Sample Input: Enter time : 12,1

Sample output : 12:01 One minute past twelve

Sample Input: Enter time : 12,45

Sample output : 12:45 quarter to one

9. Write a program to input a sentence. Pass it to different methods to carry out the corresponding tasks mentioned below:

(i) void word(String str) : to display the number of words.

(ii) void display(tring str) : to display number of vowels and uppercase characters.

(iii) void freq(String str) : to display the frequency of each character in the sentence.

Define the main() function to create an object and call the methods accordingly to enable the task.

Sample Input : Enter a String: JAVA Langage

Sample Output: No. of words : 2

No. of Vowels : 6
No. of Uppercase characters : 5

| Letters | Frequency |
|---|---|
| A | 2 |
| J | 1 |
| L | 1 |
| V | 1 |
| a | 2 |
| e | 1 |
| g | 2 |
| n | 1 |
| u | 1 |

10. A 'Sphenic Number' is a positive integer which has exactly three prime factors. The first few Sphenic numbers are : 30, 42,6,70,78,102,105,110 ,114,….so on.
Design a class Sphenic to check if a given number is a Sphenic number or not. Specify the method Check(int n) that accept a number from main method. It displays all the prime factors. If it contains only three then display 'Sphenic Number' otherwise 'Not a Sphenic Number'.
Define the main() method to create an object and call the method Check(int n) accordingly to enable the task.
Sample Input: 30
Sample Output: It's a Sphenic Number.
30= 2*3*5 . It is a product of exactly three prime numbers.
Sample Input: 60
Sample Output: It's not a Sphenic Number.
30= 2* 2 *3*5 . It is not a product of exactly three prime numbers.

11. The Consecutive prime numbers are known as Prime Triplets, if they satisfy the following condition :
(n,n+2,n+6) are all prime numbers Or (n,n+4,n+6) are all prime numbers. Where 'n' is an integer number > 0 , if n=5 then 5, 7 (5+2=7) ,11(5+6=11). Here 5,7,11 are all prime numbers so 5,7,11 are prime triplets.
If n=7, then 7, 9(7+2=9) ,11(7+6=13) but 9 s not prime.
But n=7, then 7,11(7+4=11), 13(7+6=13) are prime triplets.
Few more examples of prime triplets are:
(5,7,11), (7,11,13),(13,17,19), (17,19,23) , (37,41,43) , (41,43,47),……
Write a program to input a start limit S (>0) and a last limit L (>0). Print all prime tripets between S and L with suitable message. The prime triplets can be greater or lesser than L depending on the conditions used for generating prime number combinations. Print the total number of prime triplets at the end. Use a method that accepts three integers as parameters and returns 1 if they form a prime triplet otherwise returns 0.
Sample Input : Enter Start Limit S : 3
Enter End Limit L : 15
Sample Output :

| Prime Triplets | | |
|---|---|---|
| 5 | 7 | 11 |
| 7 | 11 | 13 |
| 13 | 17 | 19 |
| Total prime triplet combinations are : 4 | | |

Sample Input : Enter Start Limit S : 10
Enter End Limit L : 50

Sample Output :

| Prime Triplets | | |
|---|---|---|
| 11 | 13 | 17 |
| 13 | 17 | 19 |
| 17 | 19 | 23 |
| 37 | 41 | 43 |
| 41 | 43 | 47 |
| Total prime triplet combinations are : 5 | | |

12. A superclass Detail has been defined to store the details of a customer. Define a subclass Bill to compute the monthly telephone charge of the customer as per the chart is given below:
Number of calls: Rate
1 – 100: Only rental charge
101 – 200: 60 paise per call + rental charge
201 – 300: 80 paise per call + rental charge
Above 300: 1 rupee per call + rental charge
The details of both the classes are given below:
Class name: Detail
Data members/instance variables:
name: to store the name of the customer
address: to store the address of the customer
telno: to store the phone number of the customer
rent: to store the monthly rental charge
Member functions:
Detail (…): parameterized constructor to assign values to data members
void show (): to display the details of the customer
Class name: Bill
Data members/instance variables:
n: to store the number of calls
amt: to store the amount to be paid by the customer
Member functions:
Bill (…): parameterized constructor to assign values to data members of both classes and to initialize amt = 0.0
void cal(): calculate the monthly telephone charge as per the chart is given above
void show(): displays the details of the customer and amount to be paid.
Specify the class Detail giving details of the constructor, and void show(). Using the concept of inheritance, specify the class Bill giving details of the constructor(), void cal() and void show().

13. Link is an entity which can hold a maximum of 100 integers. Link enables the user to add elements from the rear end and remove integers from the front end of the entity. Define a class Link with the following details:
Class name: Link
Data Members/instance variables:
Ink []: entity to hold the integer elements,
max: stores the maximum capacity of the entity,

begin: to point to the index of the front end.
end: to point to the index of the rear end.
Member functions:
Link(intmm): constructor to initialize max = mm. begin = 0. end = 0.
void addlink (int v): to add an element from the rear index if possible otherwise display the message "OUT OF SIZE… "
int dellink(): to remove and return an element from the front index. if possible otherwise display the message "EMPTY …" and return – 99.
void display(): displays the elements of the entity.
(a) Specify the class Link giving details of the constructor (int), void addlink (int), int dellink() and void display ( ).

14. Design a Class Binary and perform the following operations as per user's choice:
    a. Add elements in the tree.
    b. Traverse the tree using PreOrder Traversal.
    c. Traverse the tree using InOrder Traversal.
    d. Traverse the tree using PostOrder Traversal.

15. A class Shift contains a two-dimensional integer array of order (m×n) where the maximum values of both m and n are 5. Design the class Shift to shuffle the matrix (i.e. the first row becomes the last, the second row becomes the first and so on). The details of the members of the class are given below:
    Class name: Shift
    Data member/instance variable:
    mat[][]: stores the array element
    m: integer to store the number of rows
    n: integer to store the number of columns
    Member functions/methods:
    Shift(int mm, intnn): parameterized constructor to initialize the data members m=mm and n=nn
    void input(): enters the elements of the array
    void cyclic(Shift p): enables the matrix of the object (P) to shift each row upwards in a cyclic manner and store the resultant matrix in the current object
    void display(): displays the matrix elements
    Specify the class Shift giving details of the constructor(), void input(), void cyclic(Shift) and void display(). Define the main() function to create an object and call the methods accordingly to enable the task of shifting the array elements.

| **S.U.P.W** |
|---|
| **Unit-I** |
| Topic : To be discussed in the class |
| **S.U.P.W** |
| Topic : To be discussed in the class |

**_____X_____**