OOP's LAB-2 and Lab-3 Experiment (Class, Object and OOP's Concepts)

Experiment 1: A department store places an order with a company for n pieces of miners, m pieces of toasters, and P number of fans. The cost of each of item is as follow:

Miners : Rs. 1,500 per piece
Toaster : Rs. 200 per piece
Fan : Rs. 450 per piece

The discount allowed for various items are 5% for miners, 15% for fan, and 10% for toaster.
The company charge 10% as sales tax on all items on net value after deducting the discount.
Write a program that reads m, n, and p and computer the amount to be paid by the store.
Note: Use the concept of inline, default argument, const function and static function.

Experiment 2:
The librarian in a library wants an application that will calculate the due date for a book given the issue date. The no. Of days in which the book is due can be decided by the librarian at the time of issued a book. For e.g. If the librarian enters the current date as 14-01-99 and the no of days in which the book is due as 15, then your program should calculate the due date and give the output as 29-01-99.
Note: Use the concept of inline, default argument, const function and static function.

Experiment 3. Design a class named Data with day, month and year (all of type int) as its data members declared in private section of the class. Define the following constructor for the class:
(i) Zero argument constructor that initialize the day, month and year data members to 12, 3 and 1993 respectively.
(ii) Parameterized constructor which takes three arguments, all of which are declared with default values as 12, 10 and 2000 respectively the day, month and year data members with corresponding arguments.
(iv) Copy constructor.

Experiment 4. Create a class named Student with the following specifications:

Private Members:

admno	of type integer to represent admission number
Name	of type character to represent name (max 20 characters)
marksEng	of type float to represent marks in the subject of English
marKsMaths	of type float to represent marks in the subject of mathematics
marksSci	of type float to represent marks in the subject of Science
total	of type float to represent total marks
compute()	A function to compute sum of marks in English, Maths and Science

Public Members:

readData()	A function to accept the data for admNo, name, marksEng, MarksMaths, marksSci. It also invokes compute function to Calculate total marks.
ShowData()	A function that display all the data member on the computer screen.

Note: Use the concept of inline, scope resolution, default argument, const function and static function.

Experiment 5: WAP o add and substract two fractions. Let the add() accept the objects using call-by-value technique, subtract() using call-by-reference, and multiply() accept using the call-by-address technique.

Experiment 6: WAP that displays the name of topper in a class.
Experiment 7: WAP that uses an overloaded constructor to dynamically allocate memory to an array and thus find the largest of its elements.

Experiment 8: Write class and object based program to deposit or withdraw money in a bank account.

OOP's LAB-2 and Lab-3 Experiment (Class, Object and OOP's Concepts)

Experiment 9: **Use OOP's concept based program to display the cheapest book available on a subject.**

Experiment 10: **Write menu-driven program that keeps a record of books and journals available in the library.**

Experiment 11: Write a program that randomly (using srand and time) displays the detail of a class period (subject and teacher).

Experiment 12: Write a program that dynamically allocates memory to a matrix. Add two matrices, display the resultant matrix, and finally free the memory space with help of destructor.
(Hint double pointer **ptr)