1. Write a program contains **Base** class that has data members: B[30] (integer), n(number of elements). It contains a function to read data member, a function to return max element in B, a function to return the factorial of max number of B (virtual function), and a function display() to display this factorial(not virtual funtion). Drive from **Base** class two subclasses **Drive1** and **Drive2**. A class **Drive1** has data member: D1[30] (large integer), a function to set the elements of D1 (each element $D1_i$ is $\sum_{j=1}^{B_i} j^2$ if B_i even or $\sum_{j=1}^{B_i} j^3$ otherwise, a function to return max element in D_1, and a function to return factorial of max element in D1. A class Drive2 has data member: D2[30] (large integer), a function to set the elements of D2 (each $D2_i$ is $\prod_{j=1}^{B_i} j^2$ if B_i even $\left(\text{or } \prod_{j=1}^{B_j} j^3 \text{ otherwise}\right)$ (if $B_i \neq 0$ put it 1), a function to return max element in D_2, and a function to return the factorial of max element in D2, Drive from two classes **Drive1** and **Drive2** a class **Drive** ,that has data member: D[30] (large integer) , a function to set data member of D (D_0 is max element in B, D_1 is max element in D1, and D_2 is the max element in D2), a function to return the factorial of the sum of all elements of D. In main function, define pointers from base class and objects from all classes to call all functions.
2. Write a program contains a template class Mat that has data members: M[30][30], n(dimension of the matrix), a function to read data members, a function to return the product of elements i^{th} column in M. Drive class Data from Mat, it contains data member: D[30], a function to set the elements of D (each element D_i is a product of elements of i^{th} column in M), a function to display all data members for Mat and Data classes in tabular form. In main function, test two objects of Data class for types: int, double, and call all function on them.