Determinant

ZPRAC-16-17-Lab12

DETERMINANT
[40 Points]
Up to 20% marks will be allotted for good programming practice.
These include
- Comments: for non-trivial code
- Indentation: align your code properly
- Function use and modular programming
- Do not include anything in the header other than what is already given in the template.
- Pointers are to be used whenever possible.
- Recursion is to be used. Other analytic methods of computing would not fetch marks.
- All matrices are to be dynamically created
- remember, you can treat matrices as an array of arrays.
0 marks otherwise.

Determinant is a very important property of square matrices. Interestingly, it can be computed in a recursive fashion as follows : Let A(n,n) be a matrix of size n x n. Let us define M(j,k) = sub-matrix of A which does not contain the row j and column k. Clearly, the matrix M would be a square matrix of size (n-1)x(n-1). We can calculate det(A) as follows : det(A) = sum over any row or column (((-1)^(j + k)) *A[j,k]* det(M(j,k)). I.e. while applying the above recursion formula, keep one of j or k fixed.

Example:
For the 3x3 matrix A we can use the above formula with j=1 fixed and varying k :-
det(A) = A[1,1]*det(M(1,1)) - A[1,2]*[det(M(1,2)) + A[1,3]*det(M(1,3)) .

(Assuming A is 1 indexed)

Input : An integer n denoting the size of the square matrix.

n lines of n integers each denoting the contents of the matrix.

Output: Determinant of the matrix. Example: Input: 2 3 1 1 6 Output: 17
Constraints: $1 \leq n \leq 5$