Before going for the case studies please go through the assignment sheet first then check the cases

7.1) Write a program to find the max, min, average, standard deviation of the elements of an integer array. ***(menu driven)***
Case studies: Since the entire syllabus is covered, try to return all the 4 parameter using one function (*remember a function can return only one parameter*).
7.2) Write a program to insert and delete an element (given by the user) into an array in a particular position (given by the user)

Case studies:
i) int insert(int A[], int n , int pos, int x)// A is the array, n is the size of the array, pos is where the insertion is going to take place.
ii) int del(int A[], int n, int pos)// do not forget to check the validity of the position!!
iii) int del(int A[], int n, int value)// this is an extra, value stands for the data, which user want to delete; there can be redundant data.

8.1) Write a program to find the transpose of a matrix.
Case studies: i) int** transpose (int row, int column, int A[] [column]); // function prototype, it is supposed to return the transpose of the matrix A. But the resultant matrix will be created dynamically.
8.2) Write a program to add two matrices
Case studies: i) int** add (int row, int column, int A[] [column], B[][column]);// before calling this function check the validity of addition, create the resultant matrix dynamically and return it.
8.3) Solve it using structure.
9.1)

Case studies:
· Compare two given strings,
int comp(char *p, char *q); // p, q are 2 pointers pointing 2 strings, returns 0 if same else returns non zero.
· Concatenate two given strings, without using standard library functions.
char* concat(char *p, char *q);// perform same as strcat(), but do check whether concatenation s at all possible for given length of strings.
· Whether a given string is palindrome or not
int palin(char * p, int len);// returns 1 if palindrome else 0

9.2) No special case studies, use string library functions to solve it.
10.1) and 10.3) no special case studies required
10.2)
Case studies: I require efficient recursive code for fibonacci series; provided that there should not be any loop to print each terms for fibonacci(n)
11.1) No special case studies required.
11.2) Write a program, using pointers, to multiply two matrices. NB: Check proper conditions for matrix multiplication.
Case studies: i) First check the validity of multiplication

Before going for the case studies please go through the assignment sheet first then check the cases

ii) int** mul(int row1, int column1, int column2, int A[][column1], int B[][column2]); // use this prototyping
12)
There is no special case study what I would like to see that you guys have used dynamic memory to store student's info. (*12.2*).