

Identify and implement boundary conditions and follow coding best practices and proper function design,

1. An array is used to maintain the rank of students in a class, the roll number of the first-rank student is stored at index 0, and the roll number of second rank student is stored at index 1 and so on. Assume the roll number is of an **integer data type**. Implement the following scenarios. There are n students in the class and the class capacity is N . $n > 5$ and $n \leq N$.

- Scenario 1:** A new student who has joined in a class, and he/she scored 3rd rank and you want to insert at index 2. Write a C program to do this task.
- Scenario 2:** A student whose rank is 1 has moved to another class (branch transfer). How does the rank list look after he/she moved? Write a C program to do this task.
- Scenario 3:** A student whose rank is 'm' has moved to another class (branch transfer). How does the rank list look after he/she moved? $m \geq 0$. Write a C program to do this task.
- Scenario 4:** A new student who has joined in a class, and he/she scored k^{th} rank and you want to insert at index $k-1$. $k \geq 1$. Write a C program to do this task.



2. A group of ten people is asked to pick an integer number and form a line(They can be in any order). Write a C program to find the people(s) with the integer 'k' and display their position(s) in the line.

3. [Additional Question] For Question 2, write an alternate logic that finds 'k' with a lesser number of comparison operations. Assume 'k' is unique(that is there no duplicates).