|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **National University of Computer and Emerging Sciences, Lahore Campus** | | | | |
| C:\Users\saif\AppData\Local\Microsoft\Windows\Temporary Internet Files\Content.Word\final design.jpg | **Course:** | **Design and Analysis of Algorithms** | **Course Code:** | **CS302** |
| **Program:** | **BS(Computer Science)** | **Semester:** | **Spring 2018** |
| **Duration:** | **10 Minutes** | **Total Marks:** | **10** |
| **Paper Date:** | **20-Feb-18** | **Weight** | **3** |
| **Section:** | **D** | **Page(s):** | **1** |
| **Exam:** | **Quiz 2** | **Roll No:** |  |
| **Section:** |  |
|  | | | | |

Suppose you are given *k* sorted arrays, each with *n* elements, and you want to combine them into a single array of *kn* elements. Consider the following approach. Using the merge subroutine taught in lecture, you merge the first 2 arrays, then merge the 3*rd* given array with this merged version of the first two arrays, then merge the 4*th* given array with the merged version of the first three arrays, and so on until you merge in the final (*kth*) input array. What is the running time taken by this successive merging algorithm, as a function of *k* and *n*? Show complete working.