

QUESTION ?

NOTE : If you take 3 minutes to answer each of the three parts to this question, that leaves you 6 minutes to read it and think. Use the 6 minutes, and give appropriately *brief* answers.

BACKGROUND.

In the notes distributed during the lecture course, there was a suggestion that random access files should be implemented as persistent arrays, declared and used in the programme as arrays of records but kept permanently on disc in the same way as files. In this question, you will explore some aspects of the feasibility of this proposal as it applies to a certain problem.

The problem concerns a structure which appears in a programme as **array example[1 .. 100000] of line**, where **line** is a structure containing two numbers **measured** and **smoothed** and other material which you don't need to worry about. One of many calculations performed on the data smoothes the **measured** values by taking a local average :

$$\text{example}[N].\text{smoothed} := (\text{example}[N-1].\text{measured} + \text{example}[N].\text{measured} + \text{example}[N+1].\text{measured}) / 3.$$

The calculations are not performed sequentially, but on demand, which may be assumed to be random.

The material in the array is stored as a **file of line**. The "other material" mentioned is just the right size to fit one record into one disc sector. (That means that there are no complications from the disc hardware when reading and writing - so you can ignore the possible complications whether or not you know about them. Each read or write operation will transfer just one record without any trouble.) By a further marvellous coincidence, the size of the disc sector is exactly the same as that of a page of memory.

THE QUESTION :

- (a) : Define "thrashing". Suppose that the problem is solved conventionally by reading the whole file into an array in memory, performing all the calculations, then writing the file back again. How could thrashing affect the performance ?
- (b) : Supposing instead that the file were implemented as a persistent array, explain what would have to happen each time an array element were used or changed.
- (c) : Compare the times taken by the smoothing operation defined above using the conventional and persistent array methods. Comment also on the total time taken to execute the programme.

NOTE : The file comes from (almost) a real example, as a part of a project which was explored but didn't materialise. The file would have contained various meteorological measurements at a measuring station at hourly intervals over a period of some years. This problem was actually three-dimensional, because measurements were to be taken by several measuring stations spread across a geographical area, so the independent variables were latitude, longitude, and time.