Practical class 8

General comments

The objectives of this eighth lab session are:

(i) To practise declaring, initialising and using arrays in C.
(ii) To learn how to write functions that take arrays as parameters.
(iii) To explore how to write functions that operate on arrays of unknown size.
(iv) To understand how arrays are effectively passed to functions by reference.
(v) To investigate what happens if we try to index an array beyond its size.

Instructions

1. First, we will write a menu system that allows the user to enter an array of numbers and then either sum the array or multiple each number in the array by a constant. To do this:

a) Declare a `double` array of size `MAX_SIZE` that will hold the user data. Choose some sensible value for `MAX_SIZE`.

b) Construct a menu system that contains three options: (i) sum an array, (ii) multiply an array by a constant, and (iii) quit the program.

c) In cases (i) and (ii), ask the user for the array size (making sure this is not larger than `MAX_SIZE`) and then read in the array values from the user. In case (ii) you will also need to ask the user for the constant multiplier.

d) Write two functions called `sum_array()` and `multiply_array()`. The first function takes two parameters (a `double` array and the size of this array) and returns the sum of the array (as a `double`). The second function takes three parameters (a `double` array, the size of this array and a multiplier) and then multiples each element of the array by the multiplier. There is no need for this function to return anything.

e) Finally, call you functions from `main()` as appropriate and print out the result (either the array sum in case (i) or the updated array in case (ii)).

2. Investigate what happens if you try to index an array beyond its declared size. Do this by creating an array of size (say) 10 and then deliberately trying to access the array beyond the tenth element. In particular, create a loop to progressively try to read your array at indices 10, 11, 12, … and so on until your program crashes. What type of crash occurs and why?

3. OPTIONAL EXTRA. Practise the use of arrays by playing John Rowe's quiz at http://newton.ex.ac.uk/teaching/resources/jmr/quiz-arrays.html.