Programming – TU856/1 & TU858/1

Lab 10 – Tuesday, December 13th, 2022

Note: You are expected to finish all programmes in your own time if you do not get these done during the lab session. This is your own responsibility.

Dynamic Memory Allocation (DMA)

Remember: Use Symbolic names in your programs. Do not hard code.

Write separate programs to:

1. Q10, Q11 (see below). These questions are not DMA but instead, practice your knowledge of using pointer notation only to interact with an array.

   Note: Be very careful with your use of pointers. Try not to exceed the bounds of the arrays, which is called buffer overflow, when moving the pointers.

2. Q9 (see below). This is your first question using DMA.

3. (i) Write a program that uses DMA to allocate memory for 5 floating-point numbers. You can use either `malloc()` or `calloc()`.

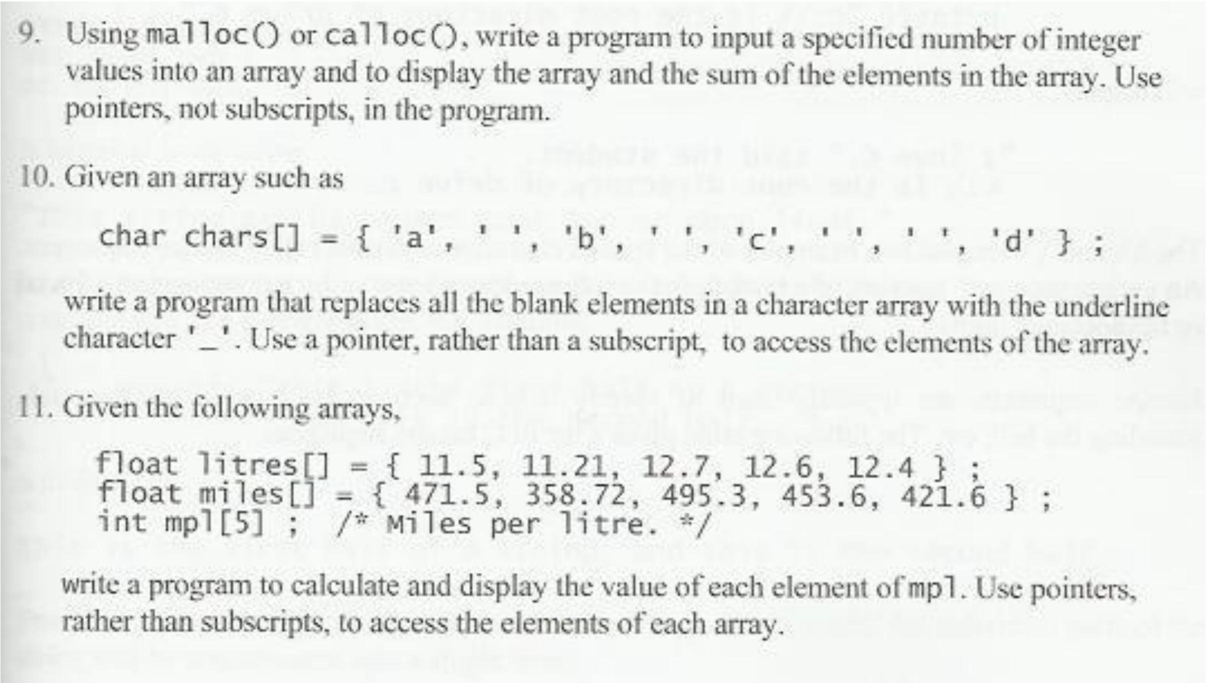
   (ii) After memory has been allocated for the 5 float numbers, enter these numbers into the memory block.

   (iii) Calculate and display the average of these numbers.

4. Create a copy of Q3 above. Modify the code so that it does:

   Careful – think first. Allocate a second memory block and store the average value calculate in part (iii) in this block. Display all of the 5 float values in the first memory block and their average value in the second memory block on the screen.

   (Hint: you will need to use 2 float pointers, one pointer to the block of memory storing the 5 floating-point numbers, the other pointer to the block of memory storing the average of the 5 numbers).



9. Using `malloc()` or `calloc()`, write a program to input a specified number of integer values into an array and to display the array and the sum of the elements in the array. Use pointers, not subscripts, in the program.

10. Given an array such as

```
char chars[] = { 'a', ' ', 'b', ' ', 'c', ' ', ' ', 'd' } ;
```

write a program that replaces all the blank elements in a character array with the underline character `'_'`. Use a pointer, rather than a subscript, to access the elements of the array.

11. Given the following arrays,

```
float litres[] = { 11.5, 11.21, 12.7, 12.6, 12.4 } ;
float miles[] = { 471.5, 358.72, 495.3, 453.6, 421.6 } ;
int mpl[5] ;  /* Miles per litre. */
```

write a program to calculate and display the value of each element of `mpl`. Use pointers, rather than subscripts, to access the elements of each array.