1806ICT Programming Fundamentals

Pointers and File IO

1. A C program contains the following declaration:

   ```
   int x[8] = {10, 20, 30, 40, 50, 60, 70, 80};
   ```

   a) What is the meaning of x?
   b) What is the meaning of x + 2?
   c) What is the value of *x?
   d) What is the value of (*x + 2)?
   e) What is the value of *(x + 2)?

2. Write a program to compute the sum of all elements in an array using pointers. The size of the array is input at run-time and the numbers are randomly generated in the range 1..20 inclusive.
3. A C program contains the following declaration:

   ```
   float table[2][3] = {{1.1, 1.2, 1.3},{2.1, 2.2, 2.3}};
   ```

   a) What is the meaning of table?
   b) What is the meaning of (table + 1)?
   c) What is the meaning of *(table + 1)?
   d) What is the meaning of (*(table + 1) + 1)?
   e) What is the meaning of *table + 1?
   f) What is the value of *(*(table + 1) + 1)?
   g) What is the value of *(*table + 1)?
   h) What is the value of *(*(table + 1))?
   i) What is the value of *(*(table) + 1) + 1?

4. The calloc(n, size) function returns a pointer to n objects (or elements), each of size size, with the memory storage initialized to zero. On the other hand, malloc(size) function returns a pointer to the requested memory with size size, but with the memory storage uninitialized.

   a. Write a program that uses the calloc() and malloc() functions to allocate memory for an array of 10 integers. Write a function void printArray(int *ptr, int n) that takes in the pointer to the array and the array size n, to print out the 10 array elements. You should see that the array created with calloc() has all its elements initialized to zero, while the array created with malloc() may not necessarily have all its elements initialized to zero.
   b. Write a function myCalloc(n, size) that uses the malloc() function to implement the functionality of calloc(). Use the printArray() function written in Part (a) above to verify that the elements of the array created with myCalloc() are all initialized to zero.

5. Write a program to encrypt the file *dictionary.txt*.
6. Write a program that reads C program source files and prints every line that contains the keywords ‘if’ or ‘for’.