Assignment #4.2

Problem: Assume there are 100 integer numbers stored in a specific SRAM area, with each integer having a value ranging from 0 to 255. Sort these numbers in ascending order using Bubble Sort.

Solution: Below showing the C code for the bubble sort.

```c
void bubble_sort() {
    int arr[100], size=100;
    for (int i = 0; i < size - 1; i++) {
        for (int j = 0; j < size - i - 1; j++) {
            if (arr[j] > arr[j + 1]) {
                int temp = arr[j];
                arr[j] = arr[j + 1];
                arr[j + 1] = temp;
            }
        }
    }
}
```

Below showing the corresponding Assembly code for the bubble sort algorithm.

```c
void bubble_sort() {
    int arr[100], size=100;          ------------------->    MOV R0,#0X20001000
                                                             MOVT R0,#0X20002000
                                                             MOV R1,#0X63
                                                             Branch1:
    for (int i = 0; i < size - 1; i++) {    ---------->      LDRB R2,#0X00
                                                             Branch2:
        for (int j = 0; j < size - i - 1; j++) {  ----->     LDRB R3,#0X00
            if (arr[j] > arr[j + 1]) {     ------------>     LDRB R4,[R0,R3]
                                                             LDRB R5,[R0,R3,#1]
                                                             CMP R4,R5
                                                             ITTT GT
                int temp = arr[j];       --------------->    STRBGT R6,R4
                arr[j] = arr[j + 1];     --------------->    STRBGT R4,R5
                arr[j + 1] = temp;       --------------->    STRBGT R5,R6
            }
        }                 ---------------------------->      ADD R3,R3,#1
                                                             CMP R3,R1
                                                             IT LT
                                                             BLT Branch2
    }       ------------------------------------------>      ADD R2,R2,#1
                                                             CMP R2,R2
```

```
23                                                  IT LT
24                                                  BLT Branch1
25  }
```