Here is a MIPS program that writes a string to an output file:

```
.data
   outfile:  .asciiz "out.txt"    # output file name
   msg:      .asciiz "This text should be written in file out.txt"
.text
   li        $v0, 13              # Service 13: open file
   la        $a0, outfile         # Output file name
   li        $a1, 1               # Write-only with create
   syscall                        # Open file
   move      $s0, $v0             # $s0 = file descriptor
   li        $v0, 15              # Service 15: write to file
   move      $a0, $s0             # $a0 = file descriptor
   la        $a1, msg             # $a1 = address of buffer
   li        $a2, 43              # $a2 = number of characters to write
   syscall                        # Write to file
   li        $v0, 16              # Service 16: close file
   move      $a0, $s0             # $a0 = file descriptor
   syscall                        # Close file
```

## 5.7 In-Lab Tasks

1. Given the following data definition statements, compute the addresses of **arr2**, **arr3**, **str1**, and **str2**, given that the address of **arr1** is **0x10010000**. Show your steps for a full mark. Select "Show Labels Window (symbol table)" from the Settings menu in MARS to check the values of your computed addresses.

   ```
   .data
   arr1: .word 5:20
   arr2: .half 7, -2, 8, -6
   arr3: .space 100
   str1: .asciiz  "This is a message"
   str2: .asciiz  "Another important string"
   ```

2. In problem 1, given that **arr1** is a one-dimensional array of integers, what are the addresses of **arr1[5]** and **arr1[17]**?

3. In problem 1, given that **arr3** is a two-dimensional array of bytes with **20** rows and **5** columns, what are the addresses of **arr3[7][2]**, **arr3[11][4]**, and **arr3[19][3]**?

4. Write a MIPS program that defines a one-dimensional array of 10 integers in the static area of the data segment, asks the user to input all 10 array elements, computes, and displays their sum.

5. Write a MIPS program that allocates an **n×n** array of integers on the heap, where **n** is a user input. The program should compute and print the value of each element as follows:

```
for (i=0; i<n; i++)
  for (j=0; j<n; j++) {
    a[i][j] = i+j;
    if (i>0) a[i][j] = a[i][j] + a[i-1][j];
    if (j>0) a[i][j] = a[i][j] + a[i][j-1];
    print_int(a[i][j]);
    print_char(' ');
  }
  print_char('\n');
}
```

6. Write a MIPS program to copy an input text file into an output file. The input and output file names should be entered by the user. If the input file cannot be opened, print an error message.

## 5.8 Bonus Question

7. Write a MIPS program to sort an array of integers in ascending order using the selection sort algorithm. The array size should be entered by the user. The array should be allocated dynamically on the heap. The array elements should be generated randomly using the random number generator. The array elements should be printed before and after sorting.