

# Computer Science 330 - Project 1

## Sorting in MIPS Assembly Language

**Due: Wed. Mar. 6, 11:59 p.m.**

The end of this handout contains a C program that reads some `int` values from the user, stores them in an array, sorts the array using the selection sort algorithm, and then prints out the sorted array. For this project, you are to implement this program in MIPS assembly language. Here are some implementation guidelines:

- Store the array in the static data area. You may assume that there will be no more than 100 elements in the array.
- Associate an `s`-register with each external variable and with the array address. You do not need to store these values in memory.
- You may use any MIPS instructions that we have covered in class, along with the pseudo-instructions `move`, `li`, `la`, and any of the four branch pseudo-instructions.

**What to turn in:** When you are ready to turn your program in, email me you assembler source code as a text file attachment. The subject of your email message should be `<Lastname> - CS 330 - Project 1`. Then, print out a copy of the source to submit in class.

```
#include <stdio.h>
#include <stdlib.h>

#define SIZE 100

int n; // Number of items in the array.
int list[SIZE]; // The array

int i, j; // loop control variables
int min_pos; // position of min element during sort
int temp; // temporary variable

int main()
{
```

```

printf ("Enter number of elements in array: ");
scanf ("%d", &n);

printf ("Enter array elements, one per line:\n");

for (i = 0; i < n; i++) {
    scanf ("%d", &list[i]);
}

// Find correct element for each position (except the last).
for (i = 0; i < n - 1; i++) {

    // Look for correct element for list[i]
    min_pos = i;
    for (j = i + 1; j < n; j++) {
        if (list[j] < list[min_pos]) {
            min_pos = j;
        }
    }

    // Swap list[min_pos] into place
    temp = list[i];
    list[i] = list[min_pos];
    list[min_pos] = temp;
}

printf ("\n");
// Print the sorted array
for (i = 0; i < n; i++) {
    printf ("%d\n", list[i]);
}

exit (0);
}

```