

Lab #1**Assigned:** 23/03/2021**Due:** 30/03/2021***Recursive Algorithm for Verifying Descending Order***

In this assignment, you are expected to write a MIPS Assembly code that checks a positive integer array and returns TRUE if the array elements are descending order; namely $A[0] > A[1] > \dots > A[n-1]$, where n is the length of the array. It returns FALSE, otherwise.

A C++-like pseudocode of the recursive algorithm is given in this document. For example, if the array is “9, 7, 6, 5, 4, 3, 2, 1”, your code must print “TRUE”. The length of the integer array is also an input of recursive algorithm. You can assume that input length is always greater than 0.

Make sure that your program complies with the MIPS conventions about register saving across procedure calls and stack usage (e.g., the caller saves argument registers while the callee saves the return address register and restores the stack pointer on exit).

You can use any pseudo-instructions you want (for instance `move` pseudo-instruction to transfer data between registers).

Notes:

1. Attached is a template (“cs401_lab1_spring2021.asm”) for your assembly program.
2. Use the MIPS simulator MARS.
3. You **HAVE TO USE** recursion in your program. Programs, which do not use recursion, will get **NO POINT**.

C++-like Pseudocode of the Recursive Algorithm for Verifying Descending Order

```
void Descending(int *A, int lenA)
{
    if (lenA <= 1)
        return TRUE
    else
        return ((A[0] > A[1]) && Descending(&A[1], lenA-1))
}
```