CSI 503 – Data Structures and Algorithms
Divide-and-Conquer Algorithm for the Max-Min Problem

[Handout 3.1]

Note: This handout is referred to on Page 3-8 of the lecture slides.

Notation and explanation: The pseudocode below uses a syntax similar to C++. The parameter A represents the given array of integers. Parameters f and l represent the indices of the first and last elements of the subarray for which the maximum and minimum need to be found (recursively). It is assumed that f ≤ l. The maximum and minimum values for the whole array A can be obtained using the call max(A, 1, n).

```c
int minmax (int A[], int f, int l) {
    if (l == f+1) { // Array size = 2. (Base case for recursion.)
        if (A[f] > A[l]) {
            max = A[f]; min = A[l]; return (min, max);
        }
        else {
            max = A[l]; min = A[f]; return (min, max);
        }
    }
    else { // Array has more than 2 elements.
        mid = (f+l)/2; // Divide step.

        (minl, maxl) = minmax(A, f, mid); // Conquer steps.
        (minr, maxr) = minmax(A, mid+1, l); // steps.

        if (maxl > maxr) // Combine steps follow.
            max = maxl;
        else max = maxr;
        if (minl < minr)
            min = minl;
        else min = minr;
        return (min, max);
    }
}
```