Readings on the Analysis of Algorithms
Fully Documented Version of the Algorithm `maxInRange`

A fully documented version of the algorithm `maxInRange` — including the precondition and postcondition for the problem being solved, a bound function for the algorithms, and assertions that document a proof of the correctness of the algorithm, is as follows.
Indeed, there is too much documentation here! For example, it is not necessary to include the fact that "`A` is an input integer array with positive length `n`" at every point. The assertions listed here are being provided because there is a ***formal method*** that can be applied, to prove that they hold, if they are given this completely. However, this is beyond the scope of these notes.

// Precondition: An integer array `A`, with positive length `n`, and integers `high` and `low`
// such that $0 \leq$ `low` $\leq$ `high` $\leq$ `n` $- 1$, are given as input.
// Postcondition: The largest element in the set
// $\{$`A[low]`, `A[low+1]`, $\ldots$, `A[high]`$\}$
// is returned as output.

```
integer maxInRange (integer[] A, integer low, integer high) {
```

// Bound Function: `high` $-$ `low`
// Assertion:
// 1. `A` is an input integer array with positive length `n`.
// 2. `low` and `high` are integer inputs such that $0 \leq$ `low` $\leq$ `high` $\leq$ `n` $- 1$.

1. `if (low == high) {`

// Assertion:
// 1. `A` is an input integer array with positive length `n`.
// 2. `low` and `high` are integer inputs such that $0 \leq$ `low` $=$ `high` $\leq$ `n` $- 1$.

2. `return A[low]`

// Assertion:
// 1. `A` is an input integer array with positive length `n`.
// 2. `low` and `high` are integer inputs such that $0 \leq$ `low` $=$ `high` $\leq$ `n` $- 1$.
// 3. The value `A[low]` $= \max($`A[low]`, `A[low+1]`, $\ldots$, `A[high]`$)$ has been returned as output.

```
} else {
```

// Assertion:
// 1. A is an input integer array with positive length n.
// 2. low and high are integer inputs such that $0 \leq$ low $<$ high $\leq$ n $- 1$.

3. integer mid := floor((low + high)/2)

// Assertion:
// 1. A is an input integer array with positive length n.
// 2. low and high are integer inputs such that $0 \leq$ low $<$ high $\leq$ n $- 1$.
// 3. mid is an integer variable such that low $\leq$ mid $\leq$ high $- 1$.

4. return max(maxInRange(A, low, mid), maxInRange(A, mid+1, high))

// Assertion:
// 1. A is an input integer array with positive length n.
// 2. low and high are integer inputs such that $0 \leq$ low $<$ high $\leq$ n $- 1$.
// 3. The value $\max($A[low], A[low+1], $\ldots,$ A[high]$)$ has been returned as output.

}

// Assertion:
// 1. A is an input integer array with positive length n.
// 2. low and high are integer inputs such that $0 \leq$ low $\leq$ high $\leq$ n $- 1$.
// 3. The value $\max($A[low], A[low+1], $\ldots,$ A[high]$)$ has been returned as output.

}