10.6.2

Proving that merge-sort is correct
Proving correctness of merge-sort

```plaintext
Merge(A[1...m], A[m + 1...n])

i ← 1, j ← m + 1, k ← 1
while (k ≤ n) do
    if i > m or (j ≤ n and A[i] > A[j])
        B[k + +] ← A[j + +]
    else
        B[k + +] ← A[i + +]
A ← B
```

Proved: Merge is correct.

```plaintext
MergeSort(A[1...n])
if n ≤ 1 then return
m ← ⌊n/2⌋
MergeSort(A[1...m])
MergeSort(A[m + 1...n])
Merge(A[1...m], A(m + 1...n))
```
Proving correctness of merge-sort

\[
\text{Merge}(A[1...m], A[m+1...n])
\]
\[
i \leftarrow 1, \ j \leftarrow m + 1, \ k \leftarrow 1
\]
\[
\text{while} \ (k \leq n) \ \text{do}
\]
\[
\quad \text{if} \ i > m \ \text{or} \ (j \leq n \ \text{and} \ A[i] > A[j])
\]
\[
\quad \quad B[k++] \leftarrow A[j++]
\]
\[
\quad \text{else}
\]
\[
\quad \quad B[k++] \leftarrow A[i++]
\]
\[
A \leftarrow B
\]

Proved: Merge is correct.

Lemma

\textbf{MergeSort} correctly sort the input array.
Proving correctness of merge-sort

Lemma

**MergeSort** correctly sort the input array.

Proof by induction on \(n\).
Proving correctness of merge-sort

Lemma

**MergeSort** correctly sort the input array.

Proof: By induction on \( n \).

```plaintext
MergeSort(A[1...n])
   if \( n \leq 1 \) then return
   m ← ⌊n/2⌋
   MergeSort(A[1...m])
   MergeSort(A[m + 1...n])
   Merge(A[1...m], A[m + 1...n])
```
Lemma

MergeSort correctly sort the input array.

Proof: By induction on \( n \).
Base: \( n = 1 \).

```
MergeSort(A[1...n])
if n ≤ 1 then return
m ← ⌊n/2⌋
MergeSort(A[1...m])
MergeSort(A[m + 1...n])
Merge(A[1...m], A(m + 1...n))
```
Lemma

**MergeSort** correctly sort the input array.

Proof: By induction on $n$.

Base: $n = 1$.

Inductive hypothesis Lemma correct for all $n \leq k$. 
Proving correctness of merge-sort

Lemma

MergeSort correctly sort the input array.

Proof: By induction on $n$.
Base: $n = 1$.
Inductive hypothesis Lemma correct for all $n \leq k$.
Inductive step: Need to prove that lemma holds for $n = k + 1 \geq 2$.
Lemma

**MergeSort** correctly sort the input array.

**Proof:** By induction on $n$.

**Base:** $n = 1$.

**Inductive hypothesis** Lemma correct for all $n \leq k$.

**Inductive step:** Need to prove that lemma holds for $n = k + 1 \geq 2$.

$m = \lfloor n/2 \rfloor < n$: Can use induction on $A[1...m]$. 

```plaintext
MergeSort(A[1...n])
if $n \leq 1$ then return
m ← $\lfloor n/2 \rfloor$
MergeSort(A[1...m])
MergeSort(A[m+1...n])
Merge(A[1...m], A(m+1...n))
```
Lemma

MergeSort correctly sort the input array.

Proof: By induction on $n$.

Base: $n = 1$.

Inductive hypothesis Lemma correct for all $n \leq k$.

Inductive step: Need to prove that lemma holds for $n = k + 1 \geq 2$.

$m = \lfloor n/2 \rfloor < n$: Can use induction on $A[1...m]$.

$n - m < n$: Can use induction on $A[m + 1...n]$. 
Proving correctness of merge-sort

Lemma

\textbf{MergeSort} correctly sort the input array.

Proof: By induction on \( n \).

Base: \( n = 1 \).

Inductive hypothesis Lemma correct for all \( n \leq k \).

Inductive step: Need to prove that lemma holds for \( n = k + 1 \geq 2 \).

\( m = \lfloor n/2 \rfloor < n \): Can use induction on \( A[1...m] \).

\( n - m < n \): Can use induction on \( A[m + 1...n] \).

\( \Rightarrow A[1...m], A[m + 1...n] \) are sorted correctly. by induction.
Proving correctness of merge-sort

Lemma

MergeSort correctly sort the input array.

Proof: By induction on $n$.

Base: $n = 1$.

Inductive hypothesis Lemma correct for all $n \leq k$.

Inductive step: Need to prove that lemma holds for $n = k + 1 \geq 2$.

$m = \lfloor n/2 \rfloor < n$: Can use induction on $A[1...m]$.

$n - m < n$: Can use induction on $A[m+1...n]$.

$\Rightarrow A[1...m], A[m+1...n]$ are sorted correctly. by induction.

Since Merge is correct $\Rightarrow A[1...n]$ is sorted correctly.
THE END

...  

(for now)