The University of Waikato
Department of Computer Science
Dr. Robi Malik

THE UNIVERSITY OF
WAIKATO
Te Whare Wānanga o Waikato

COMP 340-08B

Reasoning about Programs

Answers for Assignment 10

Exercise 1

a) The warning message I get is

```
ArrayTests.java:54: Warning: Array index possibly too large (IndexTooBig)
    int max = a[0];
              ^
```

ESC/Java2 correctly determines that it would be an error to access the first element of an empty array (an array of length 0). In this case, it is a good idea to require a non-empty array as precondition, because there is no way of finding a maximum in an empty array.

b) The conditions given correctly state that the array `a` is sorted at the end of the `bubbleSort()` procedure. But they do not mention that the contents of the array are the same (a permutation of) the input. Therefore, the program would satisfy the specification, if it returned the list $(2, 3, 4)$ when presented the input $(3, 2, 1)$.

Exercise 2

Here is my annotated program.

```
/*@
  @ requires a != null;
  @ requires a.length >= 0;
  @ ensures \result >= 0;
  @ ensures \result <= a.length;
  @ ensures (\forall int i; i >= 0 && i < \result; a[i] != item);
  @ ensures (\exists int i; i >= 0 && i < a.length; a[i] == item) ==>
  @         a[\result] == item;
  @*/
public static int sequentialSearch(final int[] a, final int item)
{
  // to prove: INIT[index/0];
  int index = 0;
  /*@ // INIT:
```

```
    @ assert index == 0;
    @*/
  // to prove: INIT ==> INVARIANT;
  /*@ // INVARIANT:
    @ assert index >= 0;
    @ assert index <= a.length;
    @ assert (\forall int i; i >= 0 && i < index; a[i] != item);
    @*/
  while (index < a.length && a[index] != item) {
    /*@ // COND:
      @ assert index < a.length && a[index] != item;
      @*/
    // to prove: INVARIANT && COND ==> INVARIANT' && COND
    /*@ // INVARIANT':
      @ assert index >= 0;
      @ assert index < a.length;
      @ assert (\forall int i; i >= 0 && i <= index; a[i] != item);
      @*/
    // to prove: INVARIANT' && COND ==> INVARIANT[index/index+1]
    index++;
    /*@ // repeat INVARIANT:
      @ assert index >= 0;
      @ assert index <= a.length;
      @ assert (\forall int i; i >= 0 && i < index; a[i] != item);
      @*/
  }
  // to prove: INVARIANT && !COND ==> POST;
  /*@ // POST:
    @ assert index >= 0;
    @ assert index <= a.length;
    @ assert (\forall int i; i >= 0 && i < index; a[i] != item);
    @ assert (\exists int i; i >= 0 && i < a.length; a[i] == item) ==>
    @        a[index] == item;
    @*/
  return index;
}
```

All the statements marked "to prove" require proving. So let us do it ...

a) $(\textbf{INIT})[\texttt{index}/0]$
$\equiv\ 0 = 0$

b) $(\textbf{INIT}) \rightarrow (\textbf{INVARIANT})$
$\equiv\ \texttt{index} = 0 \rightarrow$
$\texttt{index} \geq 0 \wedge \texttt{index} \leq \texttt{a.length} \wedge \forall i\ (i \geq 0 \wedge i < \texttt{index} \rightarrow \texttt{a}[i] \neq \texttt{item})$

Substituting the precondition $\texttt{index} = 0$ into the conclusion gives $0 \geq 0 \wedge 0 \leq \texttt{a.length} \wedge \forall i(i \geq 0 \wedge i < 0 \rightarrow \texttt{a}[i] \neq \texttt{item})$, which all is obviously true. ($0 \leq \texttt{a.length}$ must hold for every Java array, or is given by the method's precondition.)

c) $(\textbf{INVARIANT}) \wedge (\textbf{COND}) \rightarrow (\textbf{INVARIANT}') \wedge (\textbf{COND})$

$\equiv$ $\texttt{index} \geq 0 \wedge \texttt{index} \leq \texttt{a.length} \wedge$
$\forall i\ (i \geq 0 \wedge i < \texttt{index} \rightarrow \texttt{a}[i] \neq \texttt{item}) \wedge$
$\texttt{index} < \texttt{a.length} \wedge \texttt{a}[\texttt{index}] \neq \texttt{item} \rightarrow$
$\texttt{index} \geq 0 \wedge \texttt{index} < \texttt{a.length} \wedge$
$\forall i\ (i \geq 0 \wedge i < \texttt{index} \rightarrow \texttt{a}[i] \neq \texttt{item}) \wedge$
$\texttt{index} < \texttt{a.length} \wedge \texttt{a}[\texttt{index}] \neq \texttt{item}$

Since $\texttt{index} < \texttt{a.length}$ implies $\texttt{index} \leq \texttt{a.length}$, this rewrites to the obviously true formula "$A \rightarrow A$":

$\texttt{index} \geq 0 \wedge \texttt{index} < \texttt{a.length} \wedge \texttt{a}[\texttt{index}] \neq \texttt{item} \wedge$
$\forall i\ (i \geq 0 \wedge i < \texttt{index} \rightarrow \texttt{a}[i] \neq \texttt{item}) \rightarrow$
$\texttt{index} \geq 0 \wedge \texttt{index} < \texttt{a.length} \wedge \texttt{a}[\texttt{index}] \neq \texttt{item} \wedge$
$\forall i\ (i \geq 0 \wedge i < \texttt{index} \rightarrow \texttt{a}[i] \neq \texttt{item})$

d) $(\textbf{INVARIANT}') \wedge (\textbf{COND}) \rightarrow (\textbf{INVARIANT})[\texttt{index}/\texttt{index} + 1]$

$\equiv$ $\texttt{index} \geq 0 \wedge \texttt{index} < \texttt{a.length} \wedge$
$\forall i\ (i \geq 0 \wedge i < \texttt{index} \rightarrow \texttt{a}[i] \neq \texttt{item}) \wedge$
$\texttt{index} < \texttt{a.length} \wedge \texttt{a}[\texttt{index}] \neq \texttt{item} \rightarrow$
$\texttt{index} + 1 \geq 0 \wedge \texttt{index} + 1 \leq \texttt{a.length} \wedge$
$\forall i\ (i \geq 0 \wedge i < \texttt{index} + 1 \rightarrow \texttt{a}[i] \neq \texttt{item})$

In the conclusion, $\texttt{index}+1 \geq 0$ follows immediately from the precondition $\texttt{index} \geq 0$, and $\texttt{index} + 1 \leq \texttt{a.length}$ is equivalent to the precondition $\texttt{index} < \texttt{a.length}$, and the preconditions $\forall i\ (i \geq 0 \wedge i < \texttt{index} \rightarrow \texttt{a}[i] \neq \texttt{item})$ and $\texttt{a}[\texttt{index}] \neq \texttt{item}$ together imply $\forall i\ (i \geq 0 \wedge i < \texttt{index} + 1 \rightarrow \texttt{a}[i] \neq \texttt{item})$.

e) $(\textbf{INVARIANT}) \wedge \neg(\textbf{COND}) \rightarrow (\textbf{POST})$

$\equiv$ $\texttt{index} \geq 0 \wedge \texttt{index} \leq \texttt{a.length} \wedge$
$\forall i\ (i \geq 0 \wedge i < \texttt{index} \rightarrow \texttt{a}[i] \neq \texttt{item}) \wedge$
$\neg(\texttt{index} < \texttt{a.length} \wedge \texttt{a}[\texttt{index}] \neq \texttt{item}) \rightarrow$
$\texttt{index} \geq 0 \wedge \texttt{index} \leq \texttt{a.length} \wedge$
$\forall i\ (i \geq 0 \wedge i < \texttt{index} \rightarrow \texttt{a}[i] \neq \texttt{item}) \wedge$
$(\exists i\ (i \geq 0 \wedge i < \texttt{a.length} \wedge \texttt{a}[i] = \texttt{item}) \rightarrow \texttt{a}[\texttt{index}] = \texttt{item})$

The first three conjuncts in the conclusion are explicitly listed in the premise.

To prove the forth conjunct also, assume that the premises $\forall i\ (i \geq 0 \wedge i < \texttt{index} \rightarrow \texttt{a}[i] \neq \texttt{item})$ and $\neg(\texttt{index} < \texttt{a.length} \wedge \texttt{a}[\texttt{index}] \neq \texttt{item})$ are true. The assumption $\neg(\texttt{index} < \texttt{a.length} \wedge \texttt{a}[\texttt{index}] \neq \texttt{item})$ is equivalent to $\texttt{index} \geq \texttt{a.length} \vee \texttt{a}[\texttt{index}] = \texttt{item})$ by DeMorgan's law, therefore consider these two cases.

First, if $\texttt{index} \geq \texttt{a.length}$, then we have $\forall i\ (i \geq 0 \wedge i < \texttt{a.length} \rightarrow \texttt{a}[i] \neq \texttt{item})$, i.e., the precondition $\exists i\ (i \geq 0 \wedge i < \texttt{a.length} \wedge \texttt{a}[i] = \texttt{item})$ is false, proving the desired implication

$$\exists i\ (i \geq 0 \wedge i < \texttt{a.length} \wedge \texttt{a}[i] = \texttt{item}) \rightarrow \texttt{a}[\texttt{index}] = \texttt{item}$$

Second, if $\texttt{a}[\texttt{index}] = \texttt{item}$, then the conclusion of the desired implication is already true.

Exercise 3

Here is my annotated program.

```
/*@
  @ requires a != null;
  @ requires a.length >= 1;
  @ ensures (\forall int i; i >= 0 && i < a.length; a[i] <= \result);
  @ ensures (\exists int i; i >= 0 && i < a.length; a[i] == \result);
  @*/
public static int findMaximum(final int[] a)
{
  /*@ // PRE:
    @ assert a.length >= 1;
    @*/
  // to prove: PRE ==> INIT[index/1, max/a[0]];
  int index = 1;
  int max = a[0];
  /*@ // INIT:
    @ assert a.length >= 1;
    @ assert index == 1;
    @ assert max == a[0];
    @*/
  // to prove: INIT ==> INV;
  /*@ // INV:
    @ assert index <= a.length;
    @ assert (\forall int i; i >= 0 && i < index; a[i] <= max);
    @ assert (\exists int i; i >= 0 && i < index; a[i] == max);
    @*/
  while (index < a.length) {
    /*@ // WCOND:
      @ assert index < a.length;
      @*/
    if (a[index] > max) {
      /*@ // ICOND:
        @ assert a[index] > max;
        @*/
      // to prove: INV && WCOND && ICOND ==> IF1;
      /*@ // IF1:
        @ assert index < a.length;
        @ assert (\forall int i; i >= 0 && i < index; a[i] < a[index]);
        @*/
      // to prove: IF1 ==> IF2[max/a[index]];
      max = a[index];
      /*@ // IF2:
        @ assert index < a.length;
        @ assert (\forall int i; i >= 0 && i < index; a[i] < max);
        @ assert max == a[index];
        @*/
    }
    // to prove: INV && WCOND && !ICOND ==> INV1;
    // to prove: IF2 ==> INV1;
    /*@ // INV1:
```

```
    @ assert index < a.length;
    @ assert (\forall int i; i >= 0 && i <= index; a[i] <= max);
    @ assert (\exists int i; i >= 0 && i <= index; a[i] == max);
    @*/
  // to prove: INV1 ==> INV[index/index+1];
  index++;
  /*@ // repeat INV:
    @ assert index <= a.length;
    @ assert (\forall int i; i >= 0 && i < index; a[i] <= max);
    @ assert (\exists int i; i >= 0 && i < index; a[i] == max);
    @*/
 }
 // to prove: INV && !WCOND ==> POST;
 /*@ // POST:
   @ assert (\forall int i; i >= 0 && i < a.length; a[i] <= max);
   @ assert (\exists int i; i >= 0 && i < a.length; a[i] == max);
   @*/
 return max;
}
```

All the statements marked "to prove" require proving. So let us do it ...

a) $(\mathbf{PRE}) \rightarrow (\mathbf{INIT})[\texttt{index}/1, \texttt{max}/\texttt{a}[0]]$
$\equiv \quad \texttt{a.length} \geq 1 \rightarrow \texttt{a.length} \geq 1 \wedge 1 = 1 \wedge \texttt{a}[0] = \texttt{a}[0]$

b) $(\mathbf{INIT}) \rightarrow (\mathbf{INV})$
$\equiv \quad \texttt{a.length} \geq 1 \wedge \texttt{index} = 1 \wedge \texttt{max} = \texttt{a}[0] \rightarrow$
$\texttt{index} \leq \texttt{a.length} \wedge$
$\forall i \ (i \geq 0 \wedge i < \texttt{index} \rightarrow \texttt{a}[i] \leq \texttt{max}) \wedge \exists i \ (i \geq 0 \wedge i < \texttt{index} \wedge \texttt{a}[i] = \texttt{max})$

Substituting the preconditions $\texttt{index} = 1$ and $\texttt{max} = \texttt{a}[0]$ into the conclusion gives

$\texttt{a.length} \geq 1 \rightarrow$
$1 \leq \texttt{a.length} \wedge$
$\forall i \ (i \geq 0 \wedge i < 1 \rightarrow \texttt{a}[i] \leq \texttt{a}[0]) \wedge \exists i \ (i \geq 0 \wedge i < 1 \wedge \texttt{a}[i] = \texttt{a}[0])$

which is obviously true.

c) $(\mathbf{INV}) \wedge (\mathbf{WCOND}) \wedge (\mathbf{ICOND}) \rightarrow (\mathbf{IF1})$
$\equiv \quad \texttt{index} \leq \texttt{a.length} \wedge$
$\forall i \ (i \geq 0 \wedge i < \texttt{index} \rightarrow \texttt{a}[i] \leq \texttt{max}) \wedge \exists i \ (i \geq 0 \wedge i < \texttt{index} \wedge \texttt{a}[i] = \texttt{max}) \wedge$
$\texttt{index} < \texttt{a.length} \wedge \texttt{a}[\texttt{index}] > \texttt{max} \rightarrow$
$\texttt{index} < \texttt{a.length} \wedge \forall i \ (i \geq 0 \wedge i < \texttt{index} \rightarrow \texttt{a}[i] \leq \texttt{a}[\texttt{index}])$

The conclusion $\texttt{index} < \texttt{a.length}$ is listed as precondition. To see the second conclusion

$$\forall i \ (i \geq 0 \wedge i < \texttt{index} \rightarrow \texttt{a}[i] \leq \texttt{a}[\texttt{index}]) \ ,$$

let $i \geq 0$ and $i < \texttt{index}$. Then by precondition

$$\forall i \ (i \geq 0 \wedge i < \texttt{index} \rightarrow \texttt{a}[i] \leq \texttt{max})$$

it follows that $\texttt{a}[i] \leq \texttt{max}$, and together with precondition $\texttt{a}[\texttt{index}] > \texttt{max}$, this implies $\texttt{a}[i] \leq \texttt{a}[\texttt{index}]$.

d) $(\mathbf{IF1}) \rightarrow (\mathbf{IF2})[\texttt{max}/\texttt{a[index]}]$

$\equiv$ $\texttt{index} < \texttt{a.length} \wedge \forall i\ (i \geq 0 \wedge i < \texttt{index} \rightarrow \texttt{a}[i] \leq \texttt{a[index]}) \rightarrow$
$\texttt{index} < \texttt{a.length} \wedge \forall i\ (i \geq 0 \wedge i < \texttt{index} \rightarrow \texttt{a}[i] \leq \texttt{a[index]}) \wedge$
$\texttt{a[index]} = \texttt{a[index]}$

e) $(\mathbf{INV}) \wedge (\mathbf{WCOND}) \wedge \neg(\mathbf{ICOND}) \rightarrow (\mathbf{INV1})$

$\equiv$ $\texttt{index} \leq \texttt{a.length} \wedge$
$\forall i\ (i \geq 0 \wedge i < \texttt{index} \rightarrow \texttt{a}[i] \leq \texttt{max}) \wedge \exists i\ (i \geq 0 \wedge i < \texttt{index} \wedge \texttt{a}[i] = \texttt{max}) \wedge$
$\texttt{index} < \texttt{a.length} \wedge \texttt{a[index]} \leq \texttt{max} \rightarrow$
$\texttt{index} < \texttt{a.length} \wedge$
$\forall i\ (i \geq 0 \wedge i \leq \texttt{index} \rightarrow \texttt{a}[i] \leq \texttt{max}) \wedge \exists i\ (i \geq 0 \wedge i \leq \texttt{index} \wedge \texttt{a}[i] = \texttt{max}) \wedge$

The conclusion $\texttt{index} < \texttt{a.length}$ is listed as precondition. The second conclusion

$$\forall i\ (i \geq 0 \wedge i \leq \texttt{index} \rightarrow \texttt{a}[i] \leq \texttt{max})$$

follows from the preconditions

$$\forall i\ (i \geq 0 \wedge i < \texttt{index} \rightarrow \texttt{a}[i] \leq \texttt{max})$$

and $\texttt{a[index]} \leq \texttt{max}$. And the third conclusion

$$\exists i\ (i \geq 0 \wedge i \leq \texttt{index} \wedge \texttt{a}[i] = \texttt{max})$$

is implied by the precondition

$$\exists i\ (i \geq 0 \wedge i < \texttt{index} \wedge \texttt{a}[i] = \texttt{max})\ .$$

f) $(\mathbf{IF2}) \rightarrow (\mathbf{INV1})$

$\equiv$ $\texttt{index} < \texttt{a.length} \wedge \forall i\ (i \geq 0 \wedge i < \texttt{index} \rightarrow \texttt{a}[i] \leq \texttt{max}) \wedge$
$\texttt{max} = \texttt{a[index]} \rightarrow$
$\texttt{index} < \texttt{a.length} \wedge$
$\forall i\ (i \geq 0 \wedge i \leq \texttt{index} \rightarrow \texttt{a}[i] \leq \texttt{max}) \wedge \exists i\ (i \geq 0 \wedge i \leq \texttt{index} \wedge \texttt{a}[i] = \texttt{max})$

The first two conjuncts in the conclusion are explicitly listed as preconditions, and the third follows directly from the precondition $\texttt{max} = \texttt{a[index]}$.

g) $(\mathbf{INV}) \rightarrow (\mathbf{INV})[\texttt{index}/\texttt{index} + 1]$

$\equiv$ $\texttt{index} < \texttt{a.length} \wedge$
$\forall i\ (i \geq 0 \wedge i \leq \texttt{index} \rightarrow \texttt{a}[i] \leq \texttt{max}) \wedge \exists i\ (i \geq 0 \wedge i \leq \texttt{index} \wedge \texttt{a}[i] = \texttt{max}) \rightarrow$
$\texttt{index} + 1 \leq \texttt{a.length} \wedge$
$\forall i\ (i \geq 0 \wedge i < \texttt{index} + 1 \rightarrow \texttt{a}[i] \leq \texttt{max}) \wedge$
$\exists i\ (i \geq 0 \wedge i < \texttt{index} + 1 \wedge \texttt{a}[i] = \texttt{max})$

This is true because index is an integer, so $\texttt{index} < \texttt{a.length}$ is equivalent to $\texttt{index} + 1 \leq \texttt{a.length}$, and $i \leq \texttt{index}$ is equivalent to $i < \texttt{index} + 1$.

h) $(\mathbf{INV}) \wedge \neg(\mathbf{WCOND}) \rightarrow (\mathbf{POST})$

$\equiv$ $\texttt{index} \leq \texttt{a.length} \wedge$
$\forall i\ (i \geq 0 \wedge i < \texttt{index} \rightarrow \texttt{a}[i] \leq \texttt{max}) \wedge \exists i\ (i \geq 0 \wedge i < \texttt{index} \wedge \texttt{a}[i] = \texttt{max}) \wedge$
$\texttt{index} \geq \texttt{a.length} \rightarrow$
$\forall i\ (i \geq 0 \wedge i < \texttt{a.length} \rightarrow \texttt{a}[i] \leq \texttt{max}) \wedge$
$\exists i\ (i \geq 0 \wedge i < \texttt{a.length} \wedge \texttt{a}[i] = \texttt{max})$

The preconditions $\texttt{index} \leq \texttt{a.length}$ and $\texttt{index} \geq \texttt{a.length}$ together imply that $\texttt{index} = \texttt{a.length}$. Substituting this equality into the conclusion makes the conclusion equivalent to the other two preconditions.