CSCI 430: Homework 1

Garrett Hay

September 9, 2018

2.1-1

31	41	59	26	41	58
31	41	59	26	41	58
31	41	59	26	41	58
26	31	41	59	41	58
26	31	41	41	59	58
26	31	41	41	58	59

2.1-2

Algorithm 1 Reverse Sort Sort

for $j = 2$ to $A.length$ **do**
  $key = A[j]$
  $i = j - 1$
  while $i > 0$ and $A[i] < key$ **do**
    $A[i + 1] = A[i]$
    $i = i - 1$
  end while
  $A[i + 1] = key$
end for

2.1-3

Algorithm 2 Search for v

```
locate = 0
for i = 1 to A.length do
  if A[i] == v then
    locate = i
  end if
end for
if locate == 0 then
  print"NIL"
else
  print(locate)
end if
```

Initialization:

Show that 'locate', the variable that will hold the location of 'v' in the array, is 0. A number nonexistent in A[1...n].

Maintenance:

Show that the loop maintains. The body of the loop checks if A[1], A[2], and so on by 1 position to see if the current element is the same as the value of 'v' until A[n]. When an element satisfies the condition if changes 'locate' to the location in the array.

Termination:

Condition of for loop termination is that $i > A.length$ or n. Because each iterator increase by 1, we must have $i = (n + 1)$ at that time. Substituting i for $(n + 1)$ in the loop we have either 'locate' is 0 or the i were $A[i]$ is equal to v in $A[1 - n]$. Then the if-else statement decides if *locate* is 0 to print "NIL" or to print the location of v in A[]. Hence the algorithm is correct.

1 2.1-4

Formal:

Input: 2 binary numbers, A and B, represented by A[0...n] and B[0...n] in binary form
Output: Binary number $C[0...(N + 1)]$ where $C = A + B$.

Pseudo:

Algorithm 3 Binary Addition

$carry = 0$
$i = A.length$
while $i < 0$ **do**
  $added = A[i] + B[i] + carry$
  $C[i + 1] = added\%2$
  $carry = added \div 2$
  $i = i - 1$
end while
$C[0] = carry$