Question 1 **1 / 1 point**

What is the Basic Operation in the java code snippet shown below?

```
int[] v = new int[n];
for (int i = 0; i < n; i++)
    v[i] = i;
```

- comparison
- none of the above
- iteration
- ✓ assignment

Question 2 **0 / 1 point**

$\sum_{i=0}^{n} (n+1)$

What is the closed form for the summation shown above?

- $n^2 + n - 2$
- ➡ $n^2 + 2n + 1$
- $n^2 + 2n - 2$
- $n^2 - n$
- $n^2 + 2n$
- ✗ none of the above

Question 3 **1 / 1 point**

Consider the algorithm shown below. What is the output if the input is A[2,1,4,1,2]?

```
Algorithm Secret(A[0..n-1])
//Input: An array A[0..n-1] of n real numbers
minval = A[0]
maxval = A[0]
for i = 1 to n-1 do
    if A[i] < minval
       minval = A[i]
    if A[i] > maxval
       maxval = A[i]
return (maxval - minval)
```

Note: a single = is used to indicate indicate assignment in the above algo.

Answer: 3 ✓

Question 4 **1 / 1 point**

$(43 + 6!)/14 \log 2$

What is the big-oh efficiency class for the function shown above?

- $O(\log n)$
- $O(n^2)$
- $O(n \log n)$
- $O(n^3)$
- $O(2^n)$
- $O(n)$
- ✓ $O(1)$
- $O(n!)$

Question 5 **1 / 1 point**

What is the big-oh efficiency class for the algorithm shown below? Assume the input n is very large.

```
i = 0

while i < n do
  j = 0
   while j < i do
        x = x + 2
        j = j + 1
   // end of while j
    i = i + 1
// end of while i
```

- $O(n^3)$
- $O(2^n)$
- $O(n)$
- ✓ $O(n^2)$
- $O(1)$
- $O(n \log n)$
- $O(n!)$
- $O(\log n)$