Consider the following pseudocode.

```
BASE2(n)
0
      result = []
1
      i = 1
2
      while n > 0
          result[i] = n \mod 2
3
          n = |n/2|
4
          i = i + 1
5
      reverse the order of elements in result
6
7
      return result
```

- 1. Use a loop invariant for the while-loop in the pseudocode to show that the function returns an array containing the bits in the binary representation of the positive integer n, in order from most to least significant. Please remember to start your argument with a clear statement of the loop invariant. Then supply the initialization, maintenance, and termination arguments.
- 2. Give pseudocode to implement line 6 in terms of basic operations. Give a Θ -bound on the running time of your implementation. (What are you using as a measure of the size of the problem? There is a measure for which the running time depends only on the size, not on the particular instance.)
- 3. Give a Θ -bound on the running time of BASE2 as a function of n using your bound from part 2 to model the running time of line 6. You may assume that line 3 is a constant time operation. (There is a measure of the size of the problem for which T(n) does not depend on the particular instance.