Assignment 2

Quinn Perfetto, 104026025
60-454 Design and Analysis of Algorithms

February 13, 2017

Question 1 (a).

```
Algorithm 1: FlipSort(L, lower, upper)
  Input: $L[lower..upper], lower \leq i \leq upper,\ L[i] \in \{0, 1\}$
  Output: $L[lower..upper]$ sorted in ascending order
  begin
      if upper - lower > 1 then
          FlipSort(L, lower, $\lfloor \frac{lower+upper}{2} \rfloor$);
          FlipSort(L, $\lfloor \frac{lower+upper}{2} \rfloor + 1$, upper);
          return Merge($L\big[lower..\lfloor \frac{lower+upper}{2} \rfloor\big]$, $L\big[\lfloor \frac{lower+upper}{2} \rfloor + 1..upper\big]$)
```

```
Algorithm 2: Merge(A, B)
  Input: Two sorted lists $A[1..n]$ and $B[1..m]$ over $\{0, 1\}^*$
  Output: A sorted list $C[1..m + n]$ containing all elements of $A$ and $B$
  Let <> be the list concatenation operator
  begin
      $index_A := 1$;
      $index_B := 1$;
      while $A[index_A] == 0 \wedge index_A \leq n$ do
          $index_A := index_A + 1$
      while $B[index_B] == 0 \wedge index_B \leq m$ do
          $index_B := index_B + 1$
      return
      $A[1..index_A - 1] <> flip(A[index_A..n] <> B[1..index_B - 1]) <> B[index_B..m]$
```

Question 1 (b).