

# CS311 Homework 1

100 points - Due September 11

## Problem 1

Consider the following algorithm, an implementation of quicksort:

```
QUICK-SORT(list, start, end)
1: if start ≥ end then
2:   return
3: end if
4: mid ← PARTITION(list, start, end)
5: QUICK-SORT(list, start, mid)
6: QUICK-SORT(list, mid + 1, end)
```

which utilizes the implementation of partition shown below:

```
PARTITION(list, start, end)
1: left ← start + 1
2: right ← end
3: while left ≤ right do
4:   while left ≤ end ∧ list[left] < list[start] do
5:     left ← left + 1
6:   end while
7:   while right ≥ start ∧ list[right] > list[start] do
8:     right ← right - 1
9:   end while
10:  if left ≤ right then
11:    swap(list, left, right)
12:  end if
13: end while
14: swap(list, start, right)
15: return right
```

Demonstrate that this implementation is incorrect by providing an input on which the algorithm does not produce correctly sorted input and briefly explain why this input fails.

## Problem 2

Informally, an electrical circuit is a path along which electrons can flow with little or no resistance. Electrical circuits are often depicted pictorially in the manner of figure 1, which depicts an electrical circuit that allows a light to be controlled with a single switch. The switch, when thrown, either permits or prevents the flow of electrons, and when electrons flow through the light, the light turns on.

A *three-way switch*, also known as a *single-pole double-throw switch*, is a mechanical device that allows a single source of electrical current to be switched between two possible output paths. See figure 2

A *four-way switch*, also known as a *double-pole double-throw switch*, is a mechanical device that allows a pair of sources of electrical current to be switched between a pair of possible output paths. See figure 3

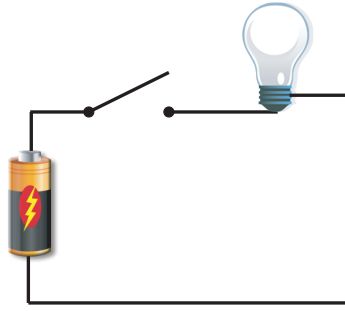


Figure 1: A light controlled by a single switch.

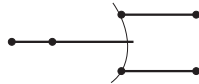


Figure 2: A single-pole double-throw switch.

- (a) Draw a wiring diagram that depicts an electrical circuit with a single light controlled by two switches (either three-way or four-way). The circuit should be configured such that throwing either switch changes the state of the light.
- (b) Draw a wiring diagram that depicts an electrical circuit with a single light controlled by three switches (each either three-way or four-way). The circuit should be configured such that throwing any switch changes the state of the light.
- (c) For each  $n > 3$ , describe an electrical circuit with a single light controlled by  $n$  switches (each either three-way or four-way). The circuit should be configured such that throwing any switch changes the state of the light.
- (d) Prove using induction that your circuits from part (c) are correct for all  $n > 3$ .

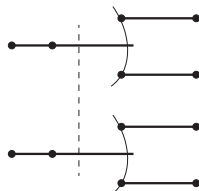


Figure 3: A double-pole double-throw switch. It can be understood to be a pair of SPDT switches mechanically connected to one another.

## Problem 3

Consider the following algorithm, a simple implementation of insertion sort:

```
INSERTION-SORT(list)
1: for  $i = 1 \rightarrow \text{list.length}$  do
2:    $v = \text{list}[i]$ 
3:    $j = i$ 
4:   while  $j > 0 \wedge \text{list}[j - 1] > v$  do
5:      $\text{list}[j] = \text{list}[j - 1]$ 
6:      $j = j - 1$ 
7:   end while
8:    $\text{list}[j] = v$ 
9: end for
```

Your objective is to prove, using two interdependent loop invariants, that this algorithm sorts the list in ascending order.

a) Identify a useful invariant  $I_o$  for the outer loop - that is, a loop invariant which will allow you to complete parts b-d.

b) Prove that if  $I_o$  holds after the last iteration of the outer loop completes then the algorithm sorts *list* in ascending order. This corresponds to the termination step of the loop invariant proof.

c) Prove that  $I_o$  holds before the loop is executed for the first time. This corresponds to the initialization step of the loop invariant proof.

d) Prove that  $I_o$  is an invariant for the outer loop. Hint: It will be necessary to prove something about the behavior of the inner loop using a loop invariant as well! This corresponds to the maintenance step of the loop invariant proof.

## Problem 4

Prove or disprove:

$$(A - B) \cup C = (A \cup B) - C$$

## Problem 5

The LONGEST COMMON SUBSEQUENCE problem asks: for some given pair of strings, what is the longest subsequence contained in both strings? Give an instance of the LONGEST COMMON SUBSEQUENCE problem.

## Problem 6

One hundred quarters lie scattered about on a table before you. Some known  $n$  of them are heads up, while the rest are tails up. You cannot in any way observe the configuration of individual quarters (perhaps you are blindfolded). You can, however, pick them up and place them back on the table in the opposite configuration, i.e., flip them.

Your objective is to divide the one hundred quarters into two groups such that each group has the same number of heads. The two groups need not have the same number of quarters. How do you do this?