15-251: Great Theoretical Ideas In Computer Science

Recitation 1 Solutions

Announcements

- **Check the course calendar** for times and locations for all course events.
- **Look at the course notes** on Diderot for definitions and example proofs.
- The **first homework assignment is out**—start early!
- Recitations and solutions will be posted online; please try to go through everything before attempting the solo problems on the homework. Extra problems are similar in difficulty to the problems we will cover, and provide more practice problems to go through. Bonus problems are a bit more difficult, but are still worth your time.
- If you have not filled out the recitation availability form, contact the professors!

Remember 76-101?

As you prepare to do the writeups next week, keep in mind the following style guidelines:

- Writing good proofs requires as much attention to the principles of English composition as to those of mathematics.
- Seriously, apply your knowledge from 76-101 when you write proofs (as much as your knowledge from 21-127/21-128/15-151).
- Write in complete sentences and pay attention to grammar. Make sure your writing is organized in coherent sections.
- Avoid run-on sentences.

Some Definitions

- Σ is your *alphabet*: non-empty and finite. Elements of Σ are called *symbols* or *characters*.
- Given an alphabet Σ, a finite *string* or *word* over Σ is a finite sequence of symbols, where each symbol is in Σ.
- Σ^* is the set of all strings over Σ of finite length, including the empty string ϵ.
- Any subset $L \subseteq \Sigma^*$ is called a *language* over Σ.
- A *computational problem* is a function $f : \Sigma^* \to \Sigma^*$.
- A *decision problem* is a function $f : \Sigma^* \to \{0, 1\}$.
- There is a one-to-one correspondence between decision problems and languages.

Note that the solutions/proofs below are bad or incorrect. Do not use them as a template.

Clearly false

Prove or disprove: for $n \in \mathbb{N}^+$, any n people all have the same hair color.

Solution: We prove the claim via induction on n.
Base case ($n = 1$): trivial.
Induction hypothesis: Suppose that for some $n \in \mathbb{N}^+$, all groups of n people have the same hair color.
Induction step: Consider a group of $n+1$ people and take the entire group except one. By the IH, these n have the same hair color. Now take another n-sized subgroup and exclude a different person. Again, these n share a hair color. Now note that the two people who were excluded both have the same hair color as the $n-1$ people who were picked twice - by transitivity, they must have the same hair color too. So all $n+1$ people have the same hair color.

Critique this solution.

The induction step skims over the fact that when $n = 2$, the set of people picked both times is empty, which breaks the proof. Proofs of false things will generally earn a grade of "Unsatisfactory".
When induction breaks, it usually breaks early on. So to sanity-check your induction proof, its a good idea to manually work through it for small cases like 1, 2, and 3. Also, lots of errors in proofs, even by professional mathematicians, can arise from trying to take an element of a set, without first proving the set is non-empty. Watch out for this.

Arrangement of ♣ and ♢

How many ways are there to arrange $c \geq 0$ ♣s and $d \geq 0$ ♢s so that all ♣s are consecutive?

Solution: You can have any number between 0 and d ♢s, then a string of ♣s; then the remainder of the ♢s. Hence, there are $d + 1$ possibilities.

Critique this solution.

This proof ignores the edge case of $c = 0$, where there is only 1 possible arrangement. Edge cases usually fall under "moderate mistakes", so this proof would be categorized as "Close". However, for some questions the edge cases are the only interesting or hard cases. Missing these cases would result in an "Unsatisfactory."

Chips in a circle

There is a circle of 15,251 chips, green on one side, red on the other. Initially, all show the green side. In one move, you may take any four consecutive chips and flip them. Is it possible to get all of the chips showing red?

Solution: No it is not possible. Let's assume for contradiction we converted all 15,251 chips to red. But this means in the very last move there must be 4 consecutive green chips and the remaining 15,247 must be red. Repeating this k times for $1 \leq k \leq 3812$, we get three consecutive red chips, with the rest green. But we started from all green, contradiction.

Critique this solution.

This solution tries to disprove the statement by assuming a particular process and then showing that the particular method won't work. It does not suffice to show that one particular method does not work. Such a solution is "unsatisfactory".

Here is a correct proof of the statement:

We wish to show that it is not possible to reach a position of 15,251 red chips in a circle, beginning from the position of 15,251 green chips in a circle, in any number of moves, where a move consists of flipping the colors of four adjacent chips. To show this, we will first prove that if we make a move from any position with an even number of red chips (and 15,251 total chips) we will leave a position with an even number of red chips.

Note that whenever we flip a chip, we either increase or decrease the number of red chips by one, and therefore the parity of the total number of red chips changes (if it was odd, it becomes even, if it was even, it becomes odd). Since we can view each of the moves from the problem description as flipping four chips in succession, each move will cause the parity to change four times, meaning that the parity of the total number of red chips will end up remaining the same after each move. Thus, we have shown that if we make a move from a position with an even number of red chips, we will leave a position with an even number of red chips.

Note that the initial position has an even number of red chips, namely 0, and the target position has an odd number of red chips, namely 15,251. Assume for the sake of contradiction that there was some sequence of moves which began with the position with 0 red chips and ended with the position with 15,251 red chips. Because this sequence of moves begins at a position with an even number of red chips and ends at a position with an odd number of red chips, there must be some move in this sequence that goes from a position with an even number of red chips to a position with an odd number of red chips. But by the result we proved in the previous paragraph, any move from a position with an even number of red chips necessarily leaves an even number of red chips, so this is impossible. We have achieved a contradiction, and thus no such sequence of moves can exist.

Balanced Parentheses

Consider the following recursively defined language L over $\{(,)\}^*$:

- $() \in L$,
- If $x \in L$, then $(x) \in L$. (WRAP)
- If $x, y \in L$, then $xy \in L$. (CONCAT)

We claim that these rules create exactly "the set of balanced strings of parentheses". But what does that even mean?

(a) How do you reasonably define a "balanced string of parentheses"?

A balanced string of parentheses must satisfy all of the following properties.

- The string is nonempty.
- The total number of open parentheses and closed parentheses are equal.
- When scanning the string from left to right, at all points in time, the number of open parentheses is at least the number of closed parentheses.

(b) Prove that a string x over $\{(,)\}^*$ is in L if and only if it satisfies your above definition of a "balanced string of parentheses".

Denote as K the language (aka set of strings) described in part a. We wish to show L = K and proceed by double containment. We break up this proof into two parts.

Part 1 : Showing $L \subseteq K$

We will prove this by structural induction on L that for any $x \in L$, $x \in K$ as well. Throughout, we will use '#open' and '#closed' as shorthand for 'number of open parentheses' and 'number of closed parentheses' respectively.

Base Case. () is a non empty string, with 1 open and closed parenthesis, and when scanning from left to right, at all points, we have #open $\geq$ #closed.

Induction step. Now, consider some string $s \in L$ that was created using either the WRAP rule or the CONCAT rule. This means that either $s = (x)$ for some $x \in L$, or $s = xy$ for some $x, y \in L$.

- **Case 1** : $s = (x)$ where $x \in L$.

  Induction Hypothesis. We assume that $x \in K$.

  Induction Step. We want to show that $s \in K$. Let's check if the three conditions for this are satisfied.

  (a) (x) is non-empty so it satisfies the first property of K

  (b) By the I.H. x ∈ K. Therefore, x has the same number of open and closed parentheses. (x) has exactly one more open parenthesis and one more closed parenthesis. Thus, (x) satisfies the second property of K.

  (c) AFSOC that at some point in time when scanning (x) from left to right, we had #open < #closed. This means that we read some closed parenthesis that made this happen. This closed parenthesis could not have been the very last closed parenthesis because at that point in time, we definitely have #open = #closed (by the previous property). This means that at some point when scanning the string "$(x$" we saw a closed parenthesis that made the third property unsatisfied. Now, suppose we scan x. When we reach that same closed parenthesis, we would again have #open < #closed (since x has fewer open parentheses than the string "$(x$"). However, this means that $x \notin K$, contradicting our induction hypothesis. Thus, s must also satisfy the third property.

- **Case 2** : $s = xy$ where $x, y \in L$.

  Induction Hypothesis. We assume that $x, y \in K$.

  Inductive Case. We want to show that $s \in K$. Let's check if the three conditions for this are satisfied.

  (a) xy is obviously nonempty (yes, you can say obviously here)

  (b) By the I.H. x ∈ K and y ∈ K implies that #open in x = #closed in x. Furthermore #open in y = #closed in y. Observe that #open in s = #open in x + #open in y. Also, #closed in s = #closed in x + #closed in y. Therefore #open in s = #closed in s.

  (c) AFSOC #open < #closed at some point during the scan. Some closed parenthesis made that happen for the first time. Either this closed parentheses is in x or in y. If it's in x the contradiction is immediate, because that means in x itself, property 3 isn't satisfied. If it's in y, there is still a contradiction because by the I.H. at the end of the string x, the value #open - #closed is 0. So, if a closed parentheses in y made xy not satisfy property 3, then it also made y not satisfy property 3. By the I.H. y ∈ K, so y must have satisfied property 3.

Part 2 : Showing $K \subseteq L$

We proceed by strong induction on the length of the string. Consider any string $s \in K$. We will show that $s \in L$ as well (i.e. it can be formed by using the rules defining L).

Base Case. Since we require a string of balanced parentheses (i.e. a string in K) to be nonempty, and since a single parenthesis can't be balanced, the smallest valid length for a string of balanced parentheses is 2. There is only one such string of length 2, namely (). Since () is defined to be in L, all length 2 strings in K are also in L, concluding the base case.

Induction Hypothesis. Suppose that for some $n \in \mathbb{N}$ (such that $n \geq 3$), all strings in K of length less than n also belong to L. We want to show that this also holds for strings in K of length n.

Induction Step. Consider some string $s \in K$ of length n. As we scan the string from left to right, let's analyze the value $v = \#$open - $\#$closed. There are two cases. Either $v \neq 0$ until the very last parenthesis, or somewhere in the middle of the string, $v = 0$.

- **Case 1** : $v \neq 0$ until the very last parenthesis of s.

  Let a be the substring between the first and last character of s. We claim that $a \in K$ (Here you will need to verify the three properties of strings in K, observe that the 3 properties are true iff v is always nonnegative in length and #open - #close = 0 at the end). Since it is of length $n-2$, by our I.H. it's also in L. Therefore, by the WRAP rule we have that $s = (a) \in L$ as desired.

- **Case 2** : v becomes 0 somewhere in the middle of s.

  Consider the string up to the point where v becomes 0. By splitting s into two strings at this point, we have two strictly shorter strings a and b. Again, by checking the 3 properties of strings in K, we can verify that indeed $a, b \in K$. By our I.H. we conclude that $a, b \in L$, and so by the CONCAT rule we have that $s = ab \in L$, completing the proof.

Inductio Ad Absurdum (Extra Problem)

It is well known that $\ln 2$ is an irrational number that is equal to the infinite sum

$$\sum_{i=1}^{\infty} \frac{(-1)^{i+1}}{i} = 1 - \frac{1}{2} + \frac{1}{3} - \frac{1}{4} + \frac{1}{5} - \dots$$

However, Leonhard claims to have a proof that shows otherwise:

He claims that $\ln 2$ is rational and will prove this by showing $\sum_{i=1}^{n} \frac{(-1)^{i+1}}{i}$ is rational for all $n > 0$ via induction.

Base Case: $n = 1$: $\sum_{i=1}^{1} \frac{(-1)^{i+1}}{i} = 1$ is indeed rational.

Induction Hypothesis: Suppose that $\sum_{i=1}^{n} \frac{(-1)^{i+1}}{i}$ is rational for $0 < n < k+1$ for some $k \in \mathbb{N}$.

Induction Step: It now suffices to show that $\sum_{i=1}^{k+1} \frac{(-1)^{i+1}}{i}$ is rational. We have that

$$\sum_{i=1}^{k+1} \frac{(-1)^{i+1}}{i} = \sum_{i=1}^{k} \frac{(-1)^{i+1}}{i} + \frac{(-1)^{k+2}}{k+1}$$

and by induction hypothesis, the first term is rational, and clearly the second term is also rational, and since the sum of two rationals is rational, we are done! ■

Where did Leonhard go wrong?

While each step of his proof is correct, it doesn't actually prove his claim! He only proved that the first n terms of the summation is rational, but not that the entire summation is rational. In other words, he didn't prove anything about the infinite sum.

Sneaky structures (Bonus Problem)

Suppose that everyone in your recitation knows at least one other person in the recitation. We say that two students are *connected* if there exists a chain of students, each consecutive pair of which know each other, spanning between the two. For example, if Xavier knows Yvonne and Yvonne knows Zachary, then Xavier and Zachary are connected even if they don't know each other. Prove or disprove: every student is connected to every other student.

Solution: We prove the claim via induction on n, the number of students.
Base case ($n = 2$): Since every student knows at least one other, the two students must know each other and are therefore connected.
Induction hypothesis: Suppose for some k that this works for all groups of k students. Induction step: Consider $k + 1$. We know the k-people recitation is connected. The $(k + 1)$th person cannot know no one, so they are connected to at least one other person in the recitation, who, by the induction hypothesis, is connected to everyone else. Thus, the whole party is still connected.

Critique this solution.

The problem here lies in the way that the induction hypothesis is applied. To apply our induction hypothesis to a group of k students, we need to know that every one of the students in that group knows at least one other person in the group. However, we only know that for the total group of size $k+1$. Suppose we pick some student s to take out. It is possible that there is a student in the remaining group of k students, who only knew s. This means that our assumption does not hold, and we cannot apply the induction hypothesis. This can be illustrated by a recitation of four people, where there are two pairs of people who know each other. There is no way to remove one student, and be left with three students where all of them know at least one of the others.
More generally, the implicit (false) assumption here is that the only way to construct graphs of minimum degree 1 is by adding one vertex at a time and connecting it to at least one previously existing vertex. (One counterexample: unions of disjoint graphs.) Correct graph induction is pretty tricky - if you're curious about how it works, feel free to ask a TA. The proper way to induct on graphs will be taught later in this course!