1 Quiz 2 Solutions

1.1 Monday 12:05-12:55 Quiz

(1) *Consider the following relations, defined on the domain D of all strings containing an arbitrary amount of as or bs ($x, y \in \{a, b\}^*$). Determine whether each relation is (i) reflexive (ii) symmetric (iii) antisymmetric (iv) transitive.*

(a) $R_1 = \{(x, y) : \exists z \in D\ s.t.\ x = yz\}$

Note that when $z = \epsilon$, $x = xz \forall x$. So the relation is reflexive. In contrast, the relation is not symmetric. This is because it is possible for a string to contain a smaller substring, but not the other way around (aba contains ab, but it is impossible for ab to contain aba due to its length). We see that the only time when xRy and yRx is possible is when x and y have the same length; and in this case, we must have $x = y$. So the relation is antisymmetric.

Finally, this relation is transitive. If xRy and yRz, this means that $\exists vy = vz$ and $\exists ux = uy$. Then, $x = uvz$, so xRz.

(b) $R_2 = \{(x, y) : xy \textit{ contains the string } bb\}$

This relation will not be reflexive; for example, the string a will not contain bb when concatenated with itself. To consider the other properties of this relation, we note that there are three (not necessarily disjoint) cases in which xy contains the string bb. They are as follows: (1) x contains bb (2) y contains bb (3) x ends in b, and y begins with b. We note that in the third case, it is possible to have xRy without yRx; a specific example of this is $ab - ba$ versus $ba - ab$. So the relation is not symmetric.

We note that an x which satifies our first case will have a relation with all other strings; that is, $\forall y xRy \wedge yRx$. So the relation is not antisymmetric.

Finally, we examine transitivity. Note that for xRy and yRz, it is possible for y to contain bb, while neither x nor z do. So the relation will not be transitive (an example of this is $a - bb, bb - aa, a - aa$.

(c) $R_3 = \{(x, y) : x \textit{ is an anagram of } y\}$ *(Ex: wasp R paws)*

We note that this relation will be an equivalence relation; only strings which contain the same characters will be anagrams of each other (and thus each such set of strings will be an equivalence class). So the relation is reflexive, symmetric, and transitive.

(2) *Prove the following statement: a simple n-vertex graph which contains only vertices of degree more than $\lceil \frac{n}{2} \rceil$ cannot be bipartite.*

We can prove the statement through contradiction; assume a bipartite graph is possible. Take an arbitrary vertex u in partition L; since u has degree more than $\lceil \frac{n}{2} \rceil$, then the partition R contains more than $\lceil \frac{n}{2} \rceil$ vertices. But each vertex in R must be connected to more than $\lceil \frac{n}{2} \rceil$ vertices; and there are less than $\frac{n}{2}$ vertices in L. Then vertices in R must be connected to each other, contradicting our assumption of a bipartite graph.

Alternately, we can see that if all vertices have degree more than $\lceil \frac{n}{2} \rceil$, the number of edges in our graph exceeds $\frac{n^2}{4}$. We can show that this is impossible using an argument included on the Wednesday 12:05-12:50 quiz.

(3) *Previously in discussion, we gave a recursive algorithm IsPath with the following specifications:*

Input: A digraph G, consisting of vertices V and an edge set E, and $u, v \in V$.
Output: True or False, depending on whether a path exists in G from u to v.

Recall that the algorithm given in class was only correct for directed acyclic graphs. We claim that IsPath can be extended to be correct for all digraphs as follows:

```
IsPath(u, v, E, V, d)
    if u = v then return true
    if d = 0 then return false
    foreach (x, y) ∈ E
        if x = u
            if IsPath(y, v, E, V, d − 1) then return true
    return false
```

where the inital call to IsPath has $d = |V| - 1$*. Modify the specification to reflect the changes to IsPath, and prove partial correctness.*

We will observe the behavior of IsPath for a few small examples to determine the significance of d. When $d = 0$, IsPath will only return true when $u = v$. When $d = 1$, IsPath will recursively call IsPath(0) on each neighbor of u; so the original call only returns true when either $u = v$ or v is a neighbor of u. More generally, we see that d represents the distance of the path from u to v; our new specification for IsPath is as follows.

Input: A digraph G, consisting of vertices V an edge set E, and $u, v \in V$, and an distance $d \in \mathbb{Z}^+$.
Output: True or False, depending on whether a path exists between u and v of length at most d.
We see that when no path exists between u and v, IsPath will return false correctly. However, we must show that whenever a path exists between u and v, IsPath will correctly return true. Our specific concern is that d will reach 0 and terminate on any path it explores before reaching v.

Let us assume a path exists between u and v. We show that this is only possible if a path of length at most $|V| - 1$ exists between u and v. Take any path of length $\geq |V|$; this path must contain some vertex w twice. We can remove all edges between the first and final occurence of w on our path; the resulting set of edges will still be a path from u to v. Repeating until no w is repeated, we necessarily have a path of length at most $|V|$.

If a path exists between u and v, either $u = v$ (the base case, which we have already shown to return correctly) or the path from u to v goes through some neighbor w of u. The edge (u, w) has length 1; so if a path of length d exists between u and v through w, then a path of length $d - 1$ exists between w and v. So IsPath$(w, v, d - 1)$ will return true, and the algorithm will be correct.

1.2 Wednesday 12:05 - 12:55 Quiz

(1) *Consider the following relations, defined on the domain D of all integers. Determine whether each relation is (i) reflexive (ii) symmetric (iii) antisymmetric (iv) transitive.*

(a) $R_1 = \{(x, y) : x < y^2\}$

We note that for $x = y = 0$, $x = y^2$. So $O \neg RO$, and the relation is not reflexive. Further, $0R1$, but $1 \neg R0$; so the relation is not symmetric. However, $2R3$ and $3R2$, so the relation is not antisymmetric either. Finally, $2R(-2)$ and $(-2)R1$, but $2 \neg R2$. So the relation is not transitive.

(b) $R_2 = \{(x, y) : |x| < 2^{|y|}\}$

This relation will be reflexive; we see that $0 < 2^0$, and then can show that $\forall x > 0, x < 2^x$ by induction. Given $n < 2^n$, we note that $2^{n+1} = 2 * 2^n$, which is greater than $n + 1$ for all n such that $2^n \geq 1$. All $n \geq 0$ satisfy this condition. So the relation is reflexive.

Note that while $0R1$, $1 \neg R0$. So the relation is not symmetric. However, due to the absolute value, we will have $(-x)Rx$ for all nonzero x. So the relation is not antisymmetric either. Finally, we note that $8R4$ and $4R3$, but $8 \neg R3$. So the relation is not transitive.

(c) $R_3 = \{(x, y) : x^2 \equiv y^2 \bmod 2\}$

Recall that we have proven in previous lectures that x is even $\iff$ x^2 is even. So the stated relation is equivalent to $x \equiv ymod2$; or 'x and y have the same parity'. This is an equivalence relation, and as such is reflexive, symmetric, and transitive. An example of $0R2$ and $2R0$ is sufficient to show that the relation is not antisymmetric.

(2) *Show that the maximum number of edges in an n-vertex bipartite graph is* $\frac{n^2}{4}$ *(assume n even).*

The maximum number of edges for a bipartite graph is certainly achieved by a fully connected graph (that is, each vertex in l is connected to each vertex in R). If $|L| = x$, then we see that $|R| = n - x$, and the total number of edges is $nx - x^2$. Setting the derivative to zero, we see that the maximum is achieved at $n = 2x$, or $x = \frac{n}{2}$. So the maximum number of edges is $\frac{n}{2}(n - \frac{n}{2}) = \frac{n^2}{4}$.

(3) *Given an arbitrary undirected graph G which can be partitioned into three sets:*

(a) *A tree C.*

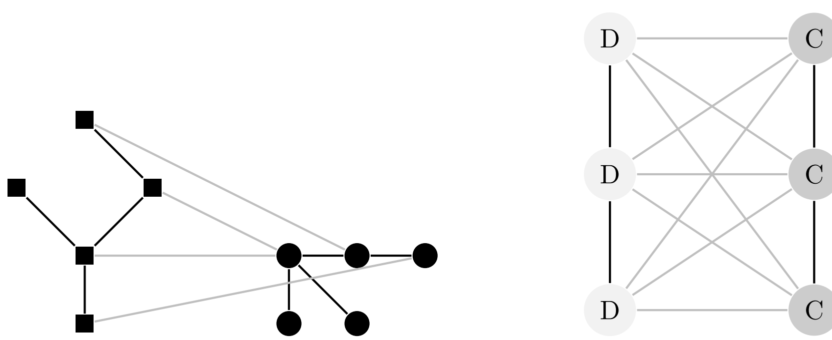
(b) *A second tree D, disjoint from C (in that it shares no edges or vertices)*

(c) *An additional set of edges E, such that* $\forall(u, v) \in E, (u \in C \land v \in D) \lor (u \in D \land v \in C)$ *[that is, each edge in E connects the two trees]*

Prove that G is four colorable. Two examples of graphs fitting the above description are provided below.

We begin by noting that any tree is two-colorable; it contains no cycles, and as such no odd-length cycles, making all trees bipartite (and thus two-colorable). So if we have four colors $1, 2, 3, 4$, we can color all vertices in tree C with $1, 2$, and D with $3, 4$. This coloring will not violate any edges in either C or D.

For all edges in E, since these edges have a vertex in each tree, and the two trees share no colors, no edge in E can have two vertices of the same color. So no edges in the larger graph G are violated, and the graph is four-colorable.



(a) G_1 (b) G_2