NAME: ______________________________

COMPSCI 250
Introduction to Computation
Second Midterm Fall 2019

M. Minea 7 November 2019

DIRECTIONS:

- Answer the problems on the exam pages.
- There are 4 problems on pages 2–6, some with multiple parts, for 100 points + 10 extra credit. Probable scale is around A=95, C=65, but will be determined after we grade the exam.
- Justify your answers and show your work. This may help with assigning partial credit.
- If you need extra space use a blank page.
- No books, notes, calculators, or collaboration.

Question 1 (20):

Let C_n be the number of strings of length n over $\Sigma = \{a, b, c\}$ that do not contain either aa or ba.
(a) Find a recurrence for C_n (i.e., a relation using previous terms of the sequence).

If a string w ends in a, it is valid precisely when the preceding letter, if any, is a c, and the prefix before it (of $n-2$ letters) is valid (has no aa or ba). If w ends in b or c, there are no constraints on any preceding letter, so the string is valid precisely if the prefix of $n-1$ letters is valid. Thus, $C_n = 2C_{n-1} + C_{n-2}$ for $n > 1$, and $C_0 = 1, C_1 = 3$ (λ and the one-letter strings are valid).

Some tried to reason about the $n-1^{\text{st}}$ letter, without realizing this may constrain both the letter before and after. There were wrong statements that $w[n-1] = a$ doubles the number of strings that can be constructed, without realizing that fixing a last letter changes the problem (counting how many of the C_n strings end with a, b or c – in fact, that's precisely C_{n-2} and C_{n-1}, for $n > 1$).

(b) Show by induction that $C_n = \big((1+\sqrt{2})^{n+1} + (1-\sqrt{2})^{n+1}\big)/2$.

Since the recurrence we found for C_n needs something other than C_{n-1}, we need *strong* induction – many did not realize that. Since C_n depends on C_{n-2}, we need to verify two base cases:
$C_0 = ((1+\sqrt{2}) + (1-\sqrt{2}))/2 = 1$, and $C_1 = ((1+\sqrt{2})^2 + (1-\sqrt{2})^2)/2 = 2 \cdot (1+2)/2 = 3$.

We assume the formula holds for all $k \leq n$, with arbitrary $n \geq 1$, and prove it for $n+1$:
$C_{n+1} = 2C_n + C_{n-1} = \frac{1}{2}\big(2(1+\sqrt{2})^{n+1} + 2(1-\sqrt{2})^{n+1} + (1+\sqrt{2})^n + (1-\sqrt{2})^n\big) =$
$\frac{1}{2}\big((1+\sqrt{2})^n(2+2\sqrt{2}+1) + (1-\sqrt{2})^n(2-2\sqrt{2}+1)\big) = \frac{1}{2}\big((1+\sqrt{2})^n(1+\sqrt{2})^2 + (1-\sqrt{2})^n(1-\sqrt{2})^2\big) =$
$\frac{1}{2}\big((1+\sqrt{2})^{n+2} + (1-\sqrt{2})^{n+2}$, so the relation holds.

A frequent error was not to verify the second base case, $C_1 = 3$. There are (infinitely) many formulas that verify the recurrence and $C_0 = 1$ but not $C_1 = 3$, for instance, $C_n = \big((1+\sqrt{2})^n + (1-\sqrt{2})^n\big)/2$. So verifying C_1 is essential for a correct proof.

Another error was to state the induction hypothesis as "for all n, $C_n = ...$". This is not induction: if we know the statement for all n, there is nothing left to prove!

Question 2 (20):

(a) Consider the sequence given by $a_0 = 0$, $a_1 = 1$, $a_n = 2a_{n-1} + a_{n-2}$ for $n > 1$.

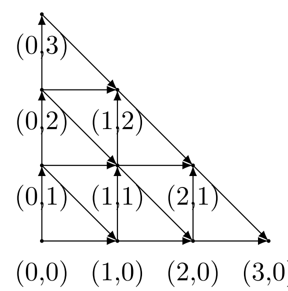
State and prove a theorem that tells for exactly which values of n the value a_n is divisible by 5.

We compute $a_2 = 2$, and for $n \geq 3$ we get $a_n = 2a_{n-1} + a_{n-2} = 2(2a_{n-2} + a_{n-3}) + a_{n-2} = 5a_{n-2} + 2a_{n-3}$, so $a_n \equiv 2a_{n-3} \pmod 5$. We claim $P(n)$: $a_n \equiv 0 \pmod 5$ if and only if $n \equiv 0 \pmod 3$. (Surprisingly many did not multiply both terms within ( ) with 2 and only got $5a_{n-2} + a_{n-3}$.)

Since our relation links a_n with a_{n-3}, we need three base cases. We check that $a_0 \equiv 0 \pmod 5$ and $a_1 = 1$, $a_2 = 2$ are not divisible by 5, so $P(0)$ through $P(2)$ hold. We use strong induction, take arbitrary $n \geq 3$, assume $P(k)$ for any $0 \leq k < n$ and prove $P(n)$. We have $a_n \equiv 2a_{n-3} \pmod 5$. Since $gcd(2, 5) = 1$, we get $a_n \equiv 0 \pmod 5 \Leftrightarrow a_{n-3} \pmod 5 \Leftrightarrow n - 3 \equiv 0 \pmod 3$ (by the IH) $\Leftrightarrow n \equiv 0 \pmod 3$, which completes the proof.

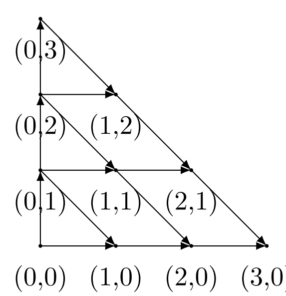
Noticing a "pattern" is not enough, we need a proof by *strong* induction. We need three base cases (0, 1, 2), since a_{n-3} is only defined for $n \geq 3$. Some writeups stated only a one-way implication (if n is divisible by 3, then a_n is divisible by 5). This is incomplete, a_n could still be divisible by 5 in other cases. A related error was to only state the property with an *if*, yet also assume in the proof that a_n is *not* divisible by 5 if n is not divisible by 3. This is wrong, our inductive step cannot assume a stronger statement than the one we're proving. The solution is to strengthen our proof goal, and with it, the inductive hypothesis, which will then make the inductive step go through.

(b) Consider the directed graph G_n (G_3 is depicted), with all edges going up, right, or down and right. More precisely, G_n has all nodes (x, y) with $0 \leq x, y \leq n$ and $x + y \leq n$, and edges $(x, y) \to (x + 1, y)$, $(x, y) \to (x, y + 1)$ and $(x, y + 1) \to (x + 1, y)$ (if both endpoints belong to G_n).
Find and prove a recurrence and then a formula for the number of directed paths from node $(0, 0)$ to the rightmost node $(n, 0)$. Justify your arguments completely and rigorously.



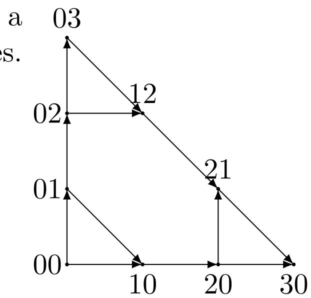
The "extra credit" problem was harder than intended, the figure had more edges than initially designed, resulting in extra path combinations. Significant partial credit was given for estimating the number of paths (many answers were around 2^n or 3^n; it can be seen that the paths more than triple after $n = 1$). Full credit was given for writing a recurrence for the number of paths reaching each node, which simply looks at where edges enter from: $P(x, 0) = P(x - 1, 0) + P(x - 1, 1)$ for $x > 0$, $P(x, y) = P(x - 1, y) + P(x, y - 1) + P(x - 1, y + 1)$ for $x, y > 0$, $P(0, y) = 1$. There is a closed form, $\sum_{k=0}^{n} \frac{1}{k+1}\binom{n}{k}\binom{n+k}{k}$, but that's not exam material, it has to do with picking k edges going up, and counting their possible orderings with respect to the corresponding k edges going right-down and the remaining $n - k$ horizontal ones.

If we remove the inner edges going up, we get a smaller and simpler count of 2^n paths.

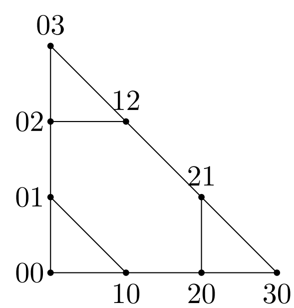


Question 3 (40p) In your graph searches, use a closed list. Show the evolution of the open list. When you need to decide which node to explore first, choose alphabetical order.

(a) In the given directed graph, carry out a DFS from node 00 without a goal node. Draw the DFS tree, and identify the type of any non-tree edges.

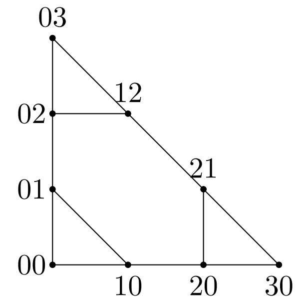


(b) In the given undirected graph, carry out a BFS from node 12 without a goal node. Draw the BFS tree, and also show any non-tree edges.

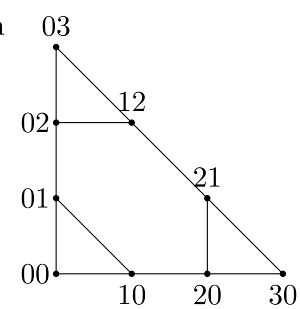


For the following two questions, the cost of diagonal edges is 1.5, all other edges have cost 1.

(c) In the given undirected graph, perform a UCS from node 03 with goal 30.



(d) In the given undirected graph, carry out an A* search from node 03 with goal node 30. The heuristic function for node xy is $h(xy) = (y + 3 - x)/2$.



Question 4 (30p)

The following are fifteen true/false questions, with no explanation needed or wanted, no partial credit for wrong answers, and no penalty for guessing.

a. The following is not a well-defined recursive function on binary strings: $f(\lambda) = 1$, $f(u0) = f(u)$, $f(u11) = f(u)$, $f(u01) = 1 - f(u)$.
TRUE, $f(1)$ is not defined.

b. If $P(0)$, $P(1)$ and $P(2)$ are true, and for all $n > 3$, $(P(n-4) \to P(n)) \vee (P(n-3) \to P(n))$ then $P(n)$ is true for all n.
FALSE, $P(5)$ could be false

c. If $P(0)$ holds, and $(P(j) \wedge P(k)) \to P(2^k(2j+1))$ for all $j, k \geq 0$, then $P(n)$ holds for all $n \geq 0$.
TRUE by strong induction, every positive natural is a product of a power of 2 with an odd number

d. Consider the relation D on naturals, so that $D(0,0)$ holds and $D(S(x), S(S(y))) \leftrightarrow D(x,y)$, where S means successor. Then $D(x,y)$ holds iff $y = 2x$.
FALSE, $D(0, Sy)$ could be anything, as well as $D(x, 1)$.

e. Let f be a function on strings, so that $f(\lambda) = \lambda$ and $f(u) = (f(u^R))^R$, where ^{R} is string reversal. Then f is the identity function.
FALSE, f is not well-defined and could be many things, including reversal

f. If nodes u and v are in different strongly connected components of a directed graph, then $P(u,v) \oplus P(v,u)$, where P is the path predicate.
FALSE, there could be no path in either direction

g. By concatenating a shortest $u \rightsquigarrow v$ path with a shortest $v \rightsquigarrow w$ path we get a shortest path $u \rightsquigarrow w$.
FALSE, consider a graph with edges $u \to v$, $v \to w$, $u \to w$.

h. For any arithmetic expression with at least two operators, either the prefix form or the postfix form contains two consecutive operators.
TRUE, there is one operator with the other as a child. Prefix if left, postfix if right.

i. If we have a sequence of n binary operators and n operands, there are at most n ways to insert another operand and make it a valid postfix expression string.
FALSE, there may be $n+1$, easily seen for $n = 1$: $a\ op \to a\ b\ op,\ b\ a\ op$

j. If an undirected graph with n nodes has a simple cycle containing all nodes, then any DFS tree will have depth $n-1$.
FALSE, consider a square with one diagonal

k. In an undirected graph, if using a closed list, the number of times a node is reached is the same in BFS and DFS from the same starting node.
TRUE, reached once from each neighbor

l. In a BFS of a directed graph, no graph edge links nodes that are more than one level apart.
FALSE, this is true for undirected graphs

m. During uniform cost search, any node u that has an edge from the start node s will be placed on the queue only once.
FALSE, it could be placed up to $n-1$ times.

n. If the heuristic h is admissible, when we take $(u, prio(u))$ off the queue, we might put on a neighbor of u with a lower value.
TRUE, we need a consistent heuristic to prevent that.

o. In a game tree with two choices at each step, which terminates in three moves (W-B-W), White might have a winning strategy even if only 2 of the 8 leaves are winning.
TRUE, example in class (one half losing, other subtrees are win/lose)