CECS 528, Pre-Final LO Assessment, Spring 2024, Dr. Ebert

NO NOTES, BOOKS, ELECTRONIC DEVICES, OR INTERPERSONAL COMMUNICATION ALLOWED. Submit each solution on a separate sheet of paper.

Problems

LO1. Complete the following problems.

(a) Compute the Jacobi symbol $\left(\frac{11}{65}\right)$.

(b) Consider the RSA key set ($N = 65 = 5 \cdot 13, e = 11$). Determine the decryption key d.

LO3. Solve each of the following problems.

(a) Recall that the `Find-Statistic` algorithm makes use of the Partitioning algorithm and uses a pivot that is guaranteed to have at least

$$3(\lfloor \frac{1}{2} \lceil \frac{n}{5} \rceil \rfloor - 2) \geq 3(\frac{1}{2} \cdot \frac{n}{5} - 3) = \frac{3n}{10} - 9.$$

members of array a both to its left and to its right. Rewrite the above inequalities but now assuming the algorithm uses groups of 13 instead of groups of five. Explain your reasoning for each of the numerical changes that you make.

(b) Demonstrate the partitioning step of Hoare's version of `Quicksort` for the array

$$a = 4, 8, 1, 6, 2, 7, 5, 3, 9, 11, 10$$

where we assume that the pivot equals the median of the first, last, and middle integers of a.

LO6. Answer the following with regards to a correctness-proof outline for Dijkstra's algorithm.

(a) In relation to Dijkstra's algorithm, provide a definition for what it means to be i) an i-neighboring path from source s to an external vertex v, and ii) the i-neighboring distance $d_i(s, v)$ from source s to external vertex v. Hint: at this point in the algorithm i nodes have been added to the DDT.

(b) Using the definitions from part a, describe the greedy choice that is made in each round of Dijkstra's algorithm.

(c) If vertex v is chosen by Dijkstra in Round $i+1$, use part b to prove that $d(s, v) = d_i(s, v)$. Hint: if i-neighboring path P from s to v has cost $d_i(s, v)$ and Q is any other path from s to v, explain why $\text{cost}(Q) \geq d_i(s, v)$.

LO7. Answer the following.

(a) Provide the dynamic-programming recurrence for computing the maximum-cost path, denoted $\mathrm{mc}(u, v)$, from a vertex u to a vertex v in a directed acyclic graph (DAG) $G = (V, E, c)$, where $c(x, y)$ gives the cost of edge $e = (x, y)$, for each $e \in E$. The recurrence should allow one to compute the maximum costs from a single source to all other vertices in a linear number of steps. Hint: step backward from v.

(b) Draw the vertices of the following DAG G in a linear left-to-right manner so that the vertices are topologically sorted, meaning, if (u, v) is an edge of G, then u appears to the left of v. The vertices of G are a-h, while the weighted edges of G are

$$(a, b, 4), (a, e, 1), (a, f, 2), (b, c, 6), (b, g, 3), (c, d, 2), (c, g, 5), (c, h, 8), (d, h, 4), (e, b, 9), (e, f, 3),$$

$$(f, b, 4), (f, c, 6), (f, g, 4), (g, d, 7), (g, h, 4).$$

(c) Starting from left to right in topological order, use the recurrence to compute

$$\mathrm{mc}(a, a), \ldots, \mathrm{mc}(a, h).$$

LO8. Consider the `2SAT` instance

$$\mathcal{C} = \{(x_1, \overline{x}_2), (x_1, \overline{x}_3), (x_1, \overline{x}_4), (\overline{x}_1, x_3), (x_2, x_4), (\overline{x}_2, x_3)(\overline{x}_3, \overline{x}_4)\}.$$

(a) Draw the implication graph $G_{\mathcal{C}}$.

(b) Find a literal l for which i) R_l is an inconsistent reachability set, ii) $R_{\overline{l}}$ is a consistent reachability set, and iii) $\alpha_{R_{\overline{l}}}$ satisfies *all* the clauses of $\mathcal{C}$. For full credit clearly state the literal l you have chosen and verify that each of the three properties are satisfied. Hint: for example, if you choose $l = \overline{x}_3$, then $\overline{l} = \overline{\overline{x}}_3 = x_3$. Justify your answer.

(c) Suppose `2SAT` instance $\mathcal{C}$ is satisfiable and the query `reachable`$(G_{\mathcal{C}}, \overline{x_3}, x_3)$ evaluates to 1. What can you say about a satisfying assignment α for $\mathcal{C}$? Defend your answer.

LO9. Answer the following.

(a) Provide the definition of what it means to be a mapping reduction from decision problem A to decision problem B.

(b) In relation to your answer to part a, if $f(n)$ is a valid mapping reduction from the `Even` decision problem to the `Odd` decision problem, then, if n is even, then what must be true about $f(n)$? Explain.

(c) Is $f(n) = n^2 + 3n + 5$ a valid mapping reduction from the `Even` decision problem to the `Odd` decision problem? Justify your answer.

LO10. An instance of the `Quadratic Diophantine` decision problem is a triple of positive integers (a, b, c), $a, b < c$, where the problem is to decide if there are positive integers x and y for which

$$ax^2 + by = c.$$

We now establish that `Quadratic Diophantine` is a member of `NP`.

(a) For a given instance (a, b, c) of `Quadratic Diophantine` describe a certificate in relation to (a, b, c).

(b) Provide a semi-formal verifier algorithm that takes as input i) an instance (a, b, c), and ii) a certificate for (a, b, c) as defined in part a, and decides if the certificate is valid for (a, b, c).

(c) Provide appropriate size parameters for `Quadratic Diophantine`. Hint: there is only one and it is neither a, nor b, nor c. Explain.

(d) Use the size parameter from part c to describe the running time of your verifier from part b. Defend your answer.

LO11. Recall the mapping reduction $f(\mathcal{C}) = (G, k)$, where f maps an instance of `3SAT` to an instance of the `Clique` decision problem. Given `3SAT` instance

$$\mathcal{C} = \{(x_1, \overline{x}_2, x_3), (\overline{x}_1, x_2, x_3), (x_1, x_2, x_3), (\overline{x}_1, \overline{x}_2, \overline{x}_3)\}$$

answer the following questions about $f(\mathcal{C})$. Hint: to answer these questions you do *not* need to draw G.

(a) How many vertices does G have? Justify your answer.

(b) How many edges does G have? Show work and justify your answer.

(c) Determine a satisfying assignment for $\mathcal{C}$ and use it to identify a k-Clique in G. Order the clique vertices so that they follow the order of the clauses of $\mathcal{C}$. Hint: there are multiple possible answers, but the clique you choose must correspond with your chosen satisfying assignment.