CECS 528, Learning Outcome Assessment 11, Spring 2024, Dr. Ebert

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

Problems

LO7. Answer/Solve the following questions/problems.

(a) The dynamic-programming algorithm that solves the `Optimal Binary Search Tree` problem defines a recurrence for the function $\text{wac}(i, j)$. In words, what does $\text{wac}(i, j)$ equal? Hint: do *not* write the recurrence (see Part b).

(b) Provide the dynamic-programming recurrence for $\text{wac}(i, j)$.

(c) Apply the recurrence from Part b to the keys 1-5 whose respective weights are 20,90,20,10,50 Show the matrix of subproblem solutions and use it to provide an optimal binary search tree. For each subproblem solution, make sure to indicate the value of k that produced the solution.

LO8. Do/answer the following.

(a) Draw the implication graph $G_{\mathcal{C}}$ associated with the `2SAT` instance

$$\mathcal{C} = \{(\overline{x}_1, x_2), (\overline{x}_1, \overline{x}_3), (x_1, x_4), (x_2, x_4), (\overline{x}_2, \overline{x}_4), (\overline{x}_2, \overline{x}_5), (\overline{x}_3, \overline{x}_4)\}.$$

(b) Apply the improved `2SAT` algorithm to obtain a satisfying assignment for $\mathcal{C}$. When deciding on the next reachability set R_l to compute, follow the literal order $l = x_1, \overline{x}_1, \ldots, x_5, \overline{x}_5$. For each consistent reachability set encountered, provide the partial assignment α_{R_l} associated with R_l and draw the reduced implication graph before continuing to the next reachability set. Note: do *not* compute the reachability set for a literal that has already been assigned a truth value. Provide a final assignment α and verify that it satisfies all the clauses.

(c) Suppose `2SAT` instance $\mathcal{C}$ is satisfiable and uses 336 variables and 615 clauses. Using the original `2SAT` algorithm, what is the *least* number of queries to a `Reachability` oracle that needs to be made in order to establish $\mathcal{C}$'s satisfiability. In other words, if we make fewer than this number of queries then it is possible that the `2SAT` instance $\mathcal{C}$ may be unsatisfiable. Explain.

LO9. Answer the following.

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

(b) An instance G of `Hamilton Path` has 563 vertices and 11294 edges. If $f : \texttt{HP} \to \texttt{LPath}$ is the mapping reduction from `Hamilton Path` to `LPath` described in lecture, then provide $f(G)$.

(c) In case G has no Hamilton path, what can we say about $f(G)$?

LO10. An instance of the `Increasing Subsequence` decision problem is an array a of n integers, and a nonnegative integer $k \leq n$. The problem is to decide if there are k indices

$$0 \leq i_1 < i_2 < \cdots < i_k < n$$

for which

$$a[i_1] < a[i_2] < \cdots < a[i_k].$$

We now establish that `Increasing Subsequence` is a member of `NP`.

(a) For a given instance (a, k) of `Increasing Subsequence` describe a certificate in relation to (a, k).

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

(c) Provide appropriate size parameters for `Increasing Subsequence`. Hint: there is only one.

(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

$$\mathcal{C} = \{(x_1, \overline{x}_2, x_5), (x_2, x_3, \overline{x}_4), (x_1, x_2, x_4), (\overline{x}_1, \overline{x}_3, \overline{x}_5)\}$$

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.