CSCI 3100 - HW04 Presentations

Group #:

Presenters:

Problem:

Kattis Success?

	3/3 points	2/3 points	1/3 poins	0/3 points
Problem decomposition	Provides a valid subproblem definition, including clear discussion of parameterization and relevant notation. Decscribes how overall problem solution is informed by subproblems.	Provides a valid subproblem definition, but with some gaps in regard to the explanation of details.	Provides a subproblem definition with significant flaws.	Unable to provide a decomposition into subproblems.
Recursive computation	Provides a clear recursive formulation that can be used to compute the "value" of one subproblem based upon zero or more other subproblems. Recursive form includes all base cases.	Provides a reasonably accurate recursive formulation, but with some errors or oversights in regard to the details of the formulation.	Provides a fundamentally flawed recursive formulation.	Shows little understanding for the relationship between subproblems.
Evaluation order of subproblems	Provides a clear explanation of how a top-down or bottom-up computation of all relevant subproblems is informed by the various dependencies among subproblems.	Provides an explanation of how top-down or bottom-up computation is used in the dynamic program, but with some minor lapses of understanding regarding the dependencies among subproblems.	Provides some discusion relating to the dependencies among subproblems, but without an actionable understanding of how top-down or bottom-up computation of values might be used.	Shows little to no understanding for how dependencies among subproblems informs a successful dynamic program.
Analysis of efficiency	Provides an accurate and clear analysis of both the running time and memory usage for a successful dynamic program, and how the resource usage relates to the overall size of a problem instance.	Provides an accurate summary of both running time and memory usage, but with less clarity in terms of the justification of one or both of those bounds, or provides clear summary of running time, but a lesser understanding of memory usage.	Provides a significantly flawed analysis of the running time of a successful dynamic program, based upon the given problem decomposition.	Shows little to no understanding for the efficiency of the proposed dynamic program, in regard to running time and/or memory.
Further exploration	Demonstrates strong understanding for how the given dynamic programming solution might (or might not) be adapted if the original problem definition were modified in unforseen ways.	Demonstrates some understanding for how the given dynamic programming solution might (or might not) be adapted if the original problem definition were modified in unforseen ways.	Struggles when asked to consider how the given dynamic programming solution might (or might not) be adapted if the original problem definition were modified in unforseen ways.	Unable to engage in meaningful conversation about how the given dynamic programming solution might (or might not) be adapted if the original problem definition were modified in unforseen ways.

Notes: