Problem Set 11-12: Complex Loops

New Learning Objectives under Evaluation

15.00 Construct and troubleshoot a flowchart using standard symbols and pseudocode

Learning Objective	Evidence
15.07 Construct a flowchart for nested structures using standard symbols and pseudocode	A decision indicating an outer structure (selection or repetition) is represented by a diamond filled with text as appropriate for the structure The outer decision has one input arrow from processes prior to the decision and two output arrows representing the Yes/No or True/False paths, and for a repetition structure one input arrow from processes inside the loop A decision indicating a nested structure (selection or repetition) is represented by a diamond filled with text as appropriate for the structure The nested decision has one input arrow from processes prior to the decision and two output arrows representing the Yes/No or True/False paths, and for a repetition structure one input arrow from processes inside the loop The Yes/No or True/False paths are clearly and appropriately labeled The arrows for each nested structure converge before returning flow to the immediate outer structure All variables used in the nested structures are initialized where appropriate Operations are included in the nested structures as required by the problem
15.08 Track a flowchart with nested structures	Describe the overall purpose of the nested structure in English Describe how the nested structure initiates and terminates in English Determine the correct number of iterations for each structure Determine the correct number of iterations for the overall nested structure Determine the value(s) of all variable(s) (re)assigned in the nested structure prior to the start of the nested structure (iteration 0) Determine the value(s) of all variable(s) (re)assigned in the nested structure at the end of each innermost structure iteration Determine the value(s) of all variable(s) (re)assigned in the nested structure at the end of the nested

17.00 Create and troubleshoot a repetition structure

Learning Objective	Evidence
17.08 Eliminate unnecessary definite looping structures	Recognize when a definite loop structure can be replaced with an element-by-element array operations Replacement of definitive loop structure with equivalent element-by-element array operations

Problem Set 11-12: Complex Loops

New Learning Objectives under Evaluation

18.00 Create and troubleshoot a repetition structure that employs vector indexing

Learning Objective	Evidence
18.01 Code an indefinite looping structure that employs vector indexing	Begin an indefinite looping structure with a `while` The `while` is followed by a condition for which a true result corresponds to code within the indefinite looping structure Variables in the condition are set correctly prior to the indefinite looping structure Variables assigned in recursive calculations within the indefinite loop are initialized correctly outside of the indefinite looping structure Vector index variables(s) are initialized outside of the indefinite looping structure Variables in the condition are updated in the indefinite loop in such a way as to make the condition false Vector index variables(s) are updated in the indefinite loop Operations within the indefinite looping structure not involving the vector index variables(s) are correct Operations within the indefinite looping structure involving vector index variables(s) (such as building or replacing values in a vector) are correct `end` is used to terminate the indefinite looping structure Statements between the `while` and `end` are indented
18.02 Track an indefinite looping structure that employs vector indexing using a variable tracking table	Describe the overall purpose of the indefinite loop that employs vector indexing in English Describe how the indefinite loop that employs vector indexing initiates and terminates in English Determine the correct number of iterations Determine the value(s) of all variable(s) (re)assigned in the indefinite loop prior to the start of the loop (iteration 0) including all values in vector variables Determine the value(s) of all variable(s) (re)assigned in the indefinite loop at the end of each iteration including all values in vector variables Determine the value(s) of all variable(s) (re)assigned in the indefinite loop at the end of loop execution including all values in vector variables
18.03 Code a definite looping structure that employs vector indexing	Begin a definite looping structure with a `for` Correct syntax for a `for` loop control statement is `for index = start_value:increment:end_value` or `for index = vector` Variables assigned in recursive calculations in the definite loop are initialized correctly outside of the definite looping structure Any vector index variable(s) **that are not part of the `for` loop control statement** is initialized outside of the definite looping structure Any vector index variable(s) **that are not part of the `for` loop control statement** is updated in the definite loop Operations within the definite looping structure not involving the vector index variables(s) are correct Operations within the definite looping structure involving vector index variable(s) (such as building or replacing values in a vector) are correct `end` is used to terminate the definite looping structure Statements between the `for` and `end` are indented

Problem Set 11-12: Complex Loops

New Learning Objectives under Evaluation

18.04 Track a definite looping structure that employs vector indexing using a variable tracking table	Describe the overall purpose of the definite loop that employs vector indexing in English Describe how the definite loop that employs vector indexing initiates and terminates in English Determine the correct number of iterations Determine the value(s) of all variable(s) (re)assigned in the definite loop prior to the start of the loop (iteration 0) including all values in vector variables Determine the value(s) of all variable(s) (re)assigned in the definite loop at the end of each iteration including all values in vector variables Determine the value(s) of all variable(s) (re)assigned in the definite loop at the end of loop execution including all values in vector variables

19.00 Create and troubleshoot nested repetition structures

Learning Objective	Evidence
19.01 Convert between these nested structures representations: English, a flowchart, and code	Correctly identify a diamond structure as a selection, indefinite looping, or definite looping structure Correctly identify the outer and inner structures Recognize variables that must be initialized before selection and indefinite structure conditional statements and operations within the nested structures Variables used in the condition and within the nested structures are initialized in the same way across representations (e.g., same values) Recognize that for indefinite and definite looping structures the Yes or True paths lead to operations within these structures Operations are completed and ordered in the same way across representations Variables are (re)assigned within the nested structure in the same way across representations (e.g., same computations) Recognize flow and symbol indicators that translate to `end` statements
19.02 Code nested structures	An appropriate outer structure is selected for the problem context An appropriate inner structure is selected for the problem context Variables are initialized as appropriate for successful execution of the nested structure Variables are updated as appropriate for successful execution of the nested structure Each structure is terminated with an `end` Indentation of the code clearly demarcates inner and outer structures

Problem Set 11-12: Complex Loops

New Learning Objectives under Evaluation

19.03 Track nested structures using a variable tracking table	Describe the overall purpose of the nested structure in English Describe how the nested structure initiates and terminates Determine the correct number of iterations for each structure Determine the correct number of iterations for the overall nested structure Determine the value(s) of all variable(s) (re)assigned in the nested structure prior to the start of the nested structure (iteration 0) Determine the value(s) of all variable(s) (re)assigned in the nested structure at the end of each innermost structure iteration Determine the value(s) of all variable(s) (re)assigned in the nested structure at the end of the nested structure execution
19.04 Code nested looping structures that employ array indexing	Variables are created to index the row and column of array(s) One repetition structure is designated to track the row index One repetition structure is designated to track the column index Array index variables are initialized as appropriate for successful execution of the nested structure Array index variables are updated as appropriate for successful execution of the nested structure Operations located within the nested structure involving the array index variables (such as building or replacing values in an array) are correct `end` is used to terminate both repetition structures Indention of the code clearly demarcates inner and outer repetition structures
19.05 Track a nested looping structure that employs array indexing using a variable tracking table	Describe the overall purpose of the nested looping structure that employs array indexing in English Describe how the nested looping structure that employs array indexing initiates and terminates Determine the correct number of iterations for the inner and outer repetition structure Determine the correct total number of iterations for the nested looping structure that employs array indexing Determine the value(s) of all variable(s) (re)assigned in nested looping structure that employs array indexing prior to the start of the loop (iteration 0) including all values in array variables Determine the value(s) of all variable(s) (re)assigned in the nested loop at the end of each innermost looping structure iteration including all values in array variables Determine the value(s) of all variable(s) (re)assigned in the nested loop at the end of loop execution including all values in array variables