Name: damian le

Date: 8/3/2025

procedure loadCourses(filename)

open file

if fail then output "Error: file not found" and return empty tree

tree ← empty BST keyed by courseNumber

for each line

tokens ← split by comma

if tokens.size < 2 then output "Bad line" and skip

courseNum ← tokens[0]

title ← tokens[1]

prereqs ← tokens[2..] (may be empty)

tree.insert(new node(courseNum, title, prereqs))

close file

// verify prereqs exist

for each node in tree

for each p in node.prereqs

if tree.search(p) not found then output "Missing prereq " + p

return tree

end

procedure insert(node, courseObj) // recursive

if node == null return new node(courseObj)

if courseObj.courseNum < node.key then node.left ← insert(node.left, courseObj)

else if courseObj.courseNum > node.key then node.right ← insert(node.right, courseObj)

else skip duplicate

return node

end

procedure printCourseList(tree)

inorder(tree.root)

end

procedure inorder(node)

if node == null return

inorder(node.left)

output node.courseNum + ": " + node.title + " Prerequisites: " + join(node.prereqs)

inorder(node.right)

end