START

OPEN CSV file

CREATE an empty Binary Tree of courseClass objects

WHILE row is not empty

GET each line of data into a string variable dataline

COUNT the number of commas in dataline string

IF number of commas = 0

PRINT error message

ELSEIF number of commas = 1

create an object, course, of courseClass

GET the first portion of the dataline string (before comma) and assign it to the classNumber field of course object

GET the second part of the dataline string (after comma) and assign it to the description field of course object

INSERT course object into binary tree

IF root is equal to null pointer

Make current bid root

ELSE

IF current bid is less than current node

Traverse left one node

IF node empty set to current bid

ELSE restart add node function

ELSEIF current bid is more than current node

Traverse right one node

IF node empty set to current bid

ELSE restart add node function

ENDIF

ENDIF

ELSEIF number of commas = 2

CREATE an object, course, of courseClass

GET the first portion of the dataline string (before comma) and assign it to the classNumber field of course object

GET the second part of the dataline string (after comma) and assign it to the description field of course object

GET the third part of the dataline string (after comma) and assign it to the prerequisite field of course object

INSERT course object into binary tree

IF root is equal to null pointer

Make current bid root

ELSE

IF current bid is less than current node

Traverse left one node

IF node empty set to current bid

ELSE restart add node function

ELSEIF current bid is more than current node

Traverse right one node

IF node empty set to current bid

ELSE restart add node function

ENDIF

ENDIF

ELSEIF number of commas = 3

CREATE an object, course, of courseClass

GET the first portion of the dataline string (before comma) and assign it to the classNumber field of course object

GET the second part of the dataline string (after comma) and assign it to the description field of course object

GET the third part of the dataline string (after comma) and assign it to the prerequisite field of course object

GET the fourth part of the dataline string (after comma) and assign it to the secondPrerequisite field of course object

INSERT course object into binary tree

IF root is equal to null pointer

Make current bid root

ELSE

IF current bid is less than current node

Traverse left one node

IF node empty set to current bid

ELSE restart add node function

ELSEIF current bid is more than current node

Traverse right one node

IF node empty set to current bid

ELSE restart add node function

ENDIF

ENDIF

ENDIF

ENDWHILE

END