

Runtime Evaluation

Vector

Code	Line Cost	# of executions	Total Cost
char ch = ''	1	1	1
String inf = ""	1	1	1
bool fileEnd = false	1	1	1
int commaCount = 0	1	1	1
int vectorPos = 0	1	1	1
fstream courseList	1	1	1
while (!fileEnd)	1	n	n
ch = fstream.getnextchar	1	n	n
if (ch == ',')	1	n	n
if (commaCount == 0)	1	n	n
courseMasterList. push_back(Course{ "", "", Vector<String>})	1	n	n
courseMasterList. at(vectorPos).courseCode = inf	1	n	n
inf = ""	1	n	n
commaCount++	1	n	n
else if (commaCount == 1)	1	n	n
courseMasterList. at(vectorPos).courseTitle = inf	1	n	n

<code>inf = ""</code>	1	n	n
<code>commaCount++</code>	1	n	n
<code>else</code>	1	n	n
<code>if (doesPrereqExist(courseMasterList, inf))</code>	1	n	n
<code>courseMasterList. at(vectorPos).pre reqs.push_back(in f)</code>	1	n	n
<code>inf = ""</code>	1	n	n
<code>commaCount++</code>	1	n	n
<code>else if (ch == newline)</code>	1	n	n
<code>if (courseMasterList .at(vectorPos).co urseCode == "" courseMasterList. at(vectorPos).cou rseTitle == "")</code>	1	n	n
<code>courseMasterList. pop_back()</code>	1	n	n
<code>commaCount = 0</code>	1	n	n
<code>else</code>	1	n	n
<code>vectorPos++</code>	1	n	n
<code>commaCount = 0</code>	1	n	n
<code>else inf = inf + ch</code>	1	n	n
<code>if (courseList.endof file == true) fileEnd = true</code>	1	1	1
Total Cost			25n+7
Runtime			O(n)

Hash Table

Code	Line Cost	# of executions	Total Cost
Vector<Course> courseMasterList	1	1	1
char ch = ''	1	1	1
String inf = ""	1	1	1
bool fileEnd = false	1	1	1
int commaCount = 0	1	1	1
int vectorPos = 0	1	1	1
fstream courseList	1	1	1
courseList.open()	1	1	1
while (!fileEnd)	1	n	n
ch = fstream.getnextch ar	1	n	n
if (ch == ',')	1	n	n
if (commaCount == 0)	1	n	n
courseMasterList. push_back(Course{ "", "", Vector<String>})	1	n	n
courseMasterList. at(vectorPos).cou rseCode = inf	1	n	n
inf = ""	1	n	n
commaCount++	1	n	n
else if (commaCount == 1)	1	n	n
courseMasterList. at(vectorPos).cou rseTitle = inf	1	n	n

<code>inf = ""</code>	1	n	n
<code>commaCount++</code>	1	n	n
<code>else</code>	1	n	n
<code>if (doesPrereqExist(courseMasterList, inf))</code>	1	n	n
<code>courseMasterList. at(vectorPos).pre reqs.push_back(in f)</code>	1	n	n
<code>inf = ""</code>	1	n	n
<code>commaCount++</code>	1	n	n
<code>else if (ch == newline)</code>	1	n	n
<code>if (courseMasterList .at(vectorPos).co urseCode == "" courseMasterList. at(vectorPos).cou rseTitle == "")</code>	1	n	n
<code>courseMasterList. pop_back()</code>	1	n	n
<code>commaCount = 0</code>	1	n	n
<code>else</code>	1	n	n
<code>vectorPos++</code>	1	n	n
<code>commaCount = 0</code>	1	n	n
<code>else inf = inf + ch</code>	1	n	n
<code>if (courseList.endof file == true)</code>	1	1	1
<code>fileEnd = true</code>	1	1	1
<code>courseList.close()</code>	1	1	1
<code>foreach(course in</code>	1	n	n

courseMasterList)			
courseHashTable.add(x)	1	n	n
courseMasterList = Vector<Course>()	1	1	1
Total Cost			27n+12
Runtime			O(n)

Binary Tree

Didn't do