Noah Figueroa
August 10, 2024
DSA Analysis and Design
Professor Adamo

Runtime Analysis

Vector - Reading the File and Creating Course Objects

Code	Line Cost	# Times Executed	Total Cost
While (getLine(inputFile, line))	1	n	n
bool found = false;	1	1	1
if (count(line.begin(), line.end(), ',') < 2)	1	1	1
cout missing parameter for course	1	1	1
exit	1	1	1
Course course;	1	1	1
istringstream stream(line);	1	1	1

getline(stream, course.courseNumber, ','); 	1	1	1
getline(stream, course.name, ','); 	1	1	1
while (getLine(stream, line)) 	1	n	n
course.addPrereq(line); 	1	n	n
found = false; 	1	1	1
for (course in courses) 	1	n^2	n^2
if (course number matches) 	1	1	1
found = true; 	1	1	1
Break; 	1	1	1
if (prereq not found as a course) 	1	1	1

cout the prerequisite course attempting to be added does not exist	1	1	1
exit	1	1	1
add course to courses vector	1	1	1

Total Cost	4N^2 + 16
Runtime	On^2 (Quadratic)

Hash Table - Reading the File and Creating Course Objects

Code	Line Cost	# Times Executed	Total Cost
while (getLine(inputFile, line))	1	n	n
bool found = false;	1	1	1
if (count(line.begin(), line.end(), ',') < 2)	1	1	1
cout missing parameter for course	1	1	1

Exit	1	1	1
Course course;	1	1	1
istringstream stream(line);	1	1	1
getline(stream, course.courseNumber, ',');	1	1	1
getline(stream, course.name, ',');	1	1	1
while (getLine(stream, line))	1	n^2	n^2
course.addPrereq(line);	1	1	1
found = false;	1	1	1
Traverse provided hash table to see if preReq exists as a course	1	n	n

if (prereq found as a course in hash table)	1	1	1
found = true;	1	1	1
Break;	1	1	1
Else if (prereq not found as a course in hash table)	1	1	1
Cout the prerequisite course attempting to be added does not exist	1	1	1
Exit	1	1	1
hashTable -> Insert(course);	1	1	1

Total Cost	N^2 * 2N + 17
Runtime	On^3 (Cubic)

Binary Search Tree - Reading the File and Creating Course Objects

Code	Line Cost	# Times Executed	Total Cost
while (getLine(inputFile, line))	1	n	n
bool found = false;	1	1	1
if (count(line.begin(), line.end(), ',') < 2)	1	1	1
cout missing parameter for course	1	1	1
Exit	1	1	1
Course course;	1	1	1
istringstream stream(line);	1	1	1
getline(stream, course.courseNumber, ',');	1	1	1
getline(stream, course.name, ',');	1	1	1
while (getLine(stream, line))	1	n	n
course.addPrereq(line);	1	1	1
found = false;	1	1	1
Traverse numbers to see if preReq exists as a course	1	n^2	n^2

If (prereq found in numbers vector)	1	1	1
Found = true;	1	1	1
Break;	1	1	1
if (prereq not found in numbers vector)	1	1	1
Cout the prerequisite course attempting to be added does not exist	1	1	1
Exit	1	1	1
tree -> Insert(course);	1	1	1

Total Cost	N^2 + 2N + 17
Runtime	On^2 (Quadratic)