**Analysis of Run-Time and Memory for Data Structures**

| **Vector** | **Line Cost** | **# Times Executes** | **Total Cost** |
| --- | --- | --- | --- |
| Open files with fstream | 1 | n | n |
| Creating a function with two parameters | 1 | n | n |
| Call to open file, WHILE file is not empty | 1 | n | n |
| Parse each line | 1 | n | n |
| IF file format is free from errors | 1 | n^2 | n^2 |
| Loop through each line in the input file | 1 | n | n |
| Parse line and extract course data | 1 | n | n |
| IF prerequisites exist, put prerequisites to OUTPUT | 1 | n | n |
| Loop through vector | 1 | n | n |
| IF INPUT equates to course number | 1 | n^2 | n^2 |
| Print prerequisites course information | 1 | 1 | 1 |
| **Total Cost** | | | 2n^2+9n + 1 |
| **Runtime** | | | O( n^2) |

| **Hash Table** | **Line Cost** | **# Times Executes** | **Total Cost** |
| --- | --- | --- | --- |
| Open files with fstream | 1 | n | n |
| Creating a function with two parameters | 1 | n | n |
| Call to open file, WHILE file is not empty | 1 | n | n |
| Parse each line | 1 | n | n |
| IF file format is free from errors | 1 | n^2 | n^2 |
| Insert method to insert items to HashTable | 1 | n | n |
| Loop through each line in the input file | 1 | n | n |
| FOR each line in file | 1 | n^2 | n^2 |
| Insert method for each value | 1 | n | n |
| IF key is found | 1 | n^2 | n^2 |
| Print out the course information | 1 | 1 | 1 |
| Print prerequisites course information | 1 | 1 | 1 |
| **Total Cost** | | | 3n^2+7n + 2 |
| **Runtime** | | | O(n^2) |

| **Tree** | **Line Cost** | **# Times Executes** | **Total Cost** |
| --- | --- | --- | --- |
| Open files with fstream | 1 | n | n |
| Creating a function with two parameters | 1 | n | n |
| Call to open file, WHILE file is not empty | 1 | n | n |
| Parse each line | 1 | n | n |
| IF file format is free from errors | 1 | n^2 | n^2 |
| Add node as parent node | 1 | 1 | 1 |
| Loop through each line in the input file | 1 | n | n |
| Initialize variables for the course | 1 | n | n |
| IF prerequisites exist | 1 | n | n |
| Assign INPUT to key | 1 | n | n |
| IF key is found | 1 | log N | log N |
| PRINT course information | 1 | 1 | 1 |
|  |  |  |  |
|  |  |  |  |
| **Total Cost** | | | log N +n^2+8n+2 |
| **Runtime** | | | O(log N) |