Brandon Lingenfelter

CS 300

05/13/2021

**Project 1**

**Pseudocode**

#include <iostream>  
#include<string>  
#include <algorithm>

using namespace std;

int main()  
{  
    cout<<"Welcome to the course Planner\n";  
    while (true)  
    {  
       int flag;  
       string flag2;  
       flag=0;  
       cout<<"\t 1.Load Data Structure.\n";  
       cout<<"\t 2.Print Course List.\n";  
       cout<<"\t 3.Print Course.\n";  
       cout<<"\t 9.Exit.\n";        
       cout<<"What would you like to do ?";  
       cin>>flag;  
       cout<<"\n";  
       if (flag==9)  
       {  
           break;  
       }  
       else if (flag==1)  
       {  
                 }  
       else if(flag==2)  
       {  
           cout<<"Here is sample schedule:\n\n";  
           cout<<"CSCI100, Introduction to Computer Science\n";  
           cout<<"CSCI101, Introduction to Programming in C++\n";  
           cout<<"CSCI200, Data Structures\n";        
           cout<<"CSCI301, Advance Programming in C++\n";  
           cout<<"CSCI300, Introduction to Algorithms\n";  
           cout<<"CSCI350, Operating Systems\n";  
           cout<<"CSCI400, Large Software Development\n";  
           cout<<"MATH201, Discrete Mathematics\n\n";  
       }  
       else if (flag==3)  
       {  
           cout<<"What course do you want to know about?";  
           cin>>flag2;  
           cout<<"\n";  
           transform(flag2.begin(), flag2.end(), flag2.begin(), ::tolower);  
           if (flag2 == "csci400")  
           {  
               cout<<"CSCI400,Large Software Development\nPrerequisites:CSCI301, CSCI350\n\n";  
           }  
            
       }  
       else  
       {  
           cout<<flag;  
           cout<<" is not a valid option\n";  
       }  
   }

    return 0;  
}

| **Code** | **Line Cost** | **# Times Executes** | **Total Cost** |
| --- | --- | --- | --- |
| **for all courses** | 1 | n | n |
| **if the course is the same as courseNumber** | 1 | n | n |
| **print out the course information** | 1 | 1 | 1 |
| **for each prerequisite of the course** | 1 | n | n |
| **print the prerequisite course information** | 1 | n | n |
| **Total Cost** | | | 4n + 1 |
| **Runtime** | | | O(n) |

The advantages to using vectors as a data structure are that they are safer at runtime as they check the type of data they contain. A disadvantage of using vectors is that data structures are objects themselves, so they can consume large portions of memory. An advantage of using binary tree structures is lookups can be done in logarithmic time which matters a lot when N is large. All the operations are guaranteed to work in O(log N) time complexity. You can implement ordered statistics and do range queries. Some disadvantages of using binary tree structures is that they need to be balanced otherwise the operations may degenerate into a linear search on an array. They can be complicated and overkill for very small number of elements. Advantages of hash tables are synchronization and they can be more efficient in many situations than other table look up structure. Some disadvantages of hash tables are that hash collisions are particularly unavoidable when hashing large sets of possible keys. Hash tables become inefficient when there are many collisions. Hash tables do not allow null values. I believe for my project I will use vectors to accomplish the goals of the project.