**5-3 Milestone: Tree Data Structure Pseudocode**

Aaron Ciminelli

Information Technology Adjunct Faculty, SNHU

CS-300-T6609 DSA: Analysis and Design 22EW6

**Wasim Alim, MSEE**

**July 31, 2022**

**Struct Course**

**String Course Name**

**String Course Number**

**Vector Prerequisite**

**Struct Node**

**Course Course**

**Create A Key For The Course**

**If Key Found**

**Return Node Course**

**If Key Is Not Found**

**Return Course**

**Class Binarysearchtree**

**Private:**

**Node Root**

**Void Add Node Node, Course**

**Public:**

**Binary Search Tree**

**Virtual ~Binarysearchtree()**

**Void In Order()**

**Void Insert Course Course**

**Course Search String Course Number**

**Binarysearchtree::Binarysearchtree**

**Root Equals Null**

**Binarysearchtree::~Binarysearchtree()**

**Void Binarysearchtree::Insert(Course Course)**

**If Equals Null**

**Root Equals New Course Node**

**Else**

**Add Node Root Course**

**Course Binarysearchtree::Search(String Coursenum)**

**Node Current Equals Root**

**If Current Is Not Null**

**If Course Number Matches**

**Return Current Course**

**If The Current Node Compare To The Course Number Is Smaller Than 0)**

**Current Equals Current Left**

**Else**

**Current Equal Right**

**Course Course**

**Return Course**

**Voidbinarysearchtree::Addnode(Node\* Node, Course Course)**

**If Node Course Number Greater Than 0 Add To Left**

**If Node Is Not Equal To Null**

**Node Becomes Left**

**Else**

**Recurse Down Left Node**

**Else**

**If No Right Node**

**Node Becomes Right**

**Else**

**Recurse Down From The Left**

**Int Total Prerequisites Equals 0**

**Int Number Of Prerequisites Courses Tree Courses String Course Num**

**Course Search Course Number**

**While Prerequisites Is Not Equal 0**

**For Prerequisites In Course Prerequisites**

**Courses Search Course Prerequisites Course Number**

**Increment Total Prequisites**

**Void Print Course Info Tree Course Number**

**Course Search Course Number**

**Print Course Information**

**Print Prerequisite Course Information**

**Course Parseline(Vector<String> $Line)**

**If Line Equals 2**

**Course New Course**

**Course Name Equals Line 0**

**Course Name Equals Line 0**

**Set Prerequisites To Empty Vector**

**Return New Course**

**Else**

**Vector Temp Prerequisites**

**For**

**Temp Prerequisities Push Back**

**Course New Course**

**Course Name Equals Line 0**

**Course Name Equals Line 0**

**Set Course Prerequisite To Temp Vector**

**Return New Course**

**Int Main()**

**Set Tree To Equal New Tree()**

**Vector Temp**

**String Line**

**Check In File File Name**

**While In Line In File**

**String Stream Line**

**While**

**String Substr**

**Get Line**

**Temp Push Back Sub Str**

**Tree Insert Parse Inline**

**Temp Clear**

**References:**

CS 300 Pseudocode word document Document

Vahid , F., Lysecky, S., Wheatland, N., Siu, R., Lysecky, R., Edgcomb, A., & Yuen, J. (2019). *CS 300: Data Structures and Algorthims*. Zybooks. Retrieved July 31, 2022, from https://learn.zybooks.com/zybook/CS-300-T6609-OL-TRAD-UG.22EW6/chapter/6/section/1