Data Structures- Assignment 1

For: Dr. Ramazan Aygün

By: Erik Failing

2/4/2019

Table of Contents

Section I.1. System Overview

This program handles the reading, organization and output of users in some database.

Section I.1.1. Referenced Documents

No references were made.

Section I.1.2. Concept of Execution

A database of users will be read in from a text file and be outputted to the console and another text file.

Section I.1.3. Abstract Data Type

Not applicable, no custom ADT was used.

Section I.2. Code Outline

Class: User

void Initialize(const char* afName, const char* alName, char aGender, const char* aMajor, const char* aEmail, AddressType aAddress, float aGPA, DateType aDateOfBirth);

void GetFirstName(char afname[]) const;

void GetLastName(char alname[]) const;

void GetMajor(char amajor[]) const;

void GetEmail(char aemail[]) const;

void GetGender(char& agender) const;

DateType GetDateOfBirth() const;

void GetDateOfBirth(DateType& aDateOfBirth) const;

float GetGPA() const;

void GetGPA(float& aGPA) const;

AddressType GetAddress() const;

void GetAddress(AddressType& aAddress) const;

void GetAddress(char aStreetName[], int& aStreetNo, char aCity[], int& aZip, char aState[]) const;

void SetFirstName(const char* afName);

void SetLastName(const char* alName);

void SetMajor(const char* amajor);

void SetEmail(const char* anemail);

void SetGender(char aGender);

void SetDateOfBirth(DateType aDateOfBirth);

void SetGPA(float aGPA);

void SetAddress(AddressType aAddress);

void SetAddress(char aStreetName[], int aStreetNo, char aCity[], int aZip, char aState[]);

void Display() const;

void Display(std::ofstream& outFile) const;

Class: Test Driver

int Populate(const char* input, User users[]);

void Test(User users[], int count) const;

void Test(std::ofstream& outFile, User users[], int count) const;

Section I.2.1. Detailed Design

Class: User

void Initialize(const char* afName, const char* alName, char aGender, const char* aMajor, const char* aEmail, AddressType aAddress, float aGPA, DateType aDateOfBirth);

Purpose: Creates a new user

Arguments: Char arrays for the user's first name, last name, major and email. A float for GPA. An AddressType for the user's address and DateType for the user's date of birth.

Return Value: Void

void GetFirstName(char afname[]) const;

Purpose: Gets the user's first name

Arguments: A character array that is the user's first name

Return Value: Void

void GetLastName(char alname[]) const;

Purpose: Gets the user's last name.

Arguments: A character array that is the user's last name

Return Value: Void

void GetMajor(char amajor[]) const;

Purpose: Gets the user's major

Arguments: A char array that is the user's major

Return Value: Void

void GetEmail(char aemail[]) const;

Purpose: Gets the user's email

Arguments: A char array that is the user's email

Return Value:

void GetGender(char& agender) const;

Purpose: Gets the user's gender

Arguments: A char that is the user's gender

Return Value: Void

DateType GetDateOfBirth() const;

Purpose: Gets the user's date of birth

Arguments: None

Return Value: Returns a DateType which is three integers representing the user's date of birth

void GetDateOfBirth(DateType& aDateOfBirth) const;

Purpose: Gets the user's date of birth

Arguments: DateType which is three integers representing the user's date of birth

Return Value: Void

float GetGPA() const;

Purpose: Gets the user's GPA

Arguments: None

Return Value: Returns a float which is the user's GPA

void GetGPA(float& aGPA) const;

Purpose: Gets the user's GPA

Arguments: A float which is the user's GPA

Return Value: Void

AddressType GetAddress() const;

Purpose: Gets the user's address

Arguments: None

Return Value: Returns an AddressType which is three char arrays and two integers making up the user's address

void GetAddress(AddressType& aAddress) const;

Purpose: Gets the user's address

Arguments: An AddressType which is three char arrays and two integers making up the user's address

Return Value: Void

void GetAddress(char aStreetName[], int& aStreetNo, char aCity[], int& aZip, char aState[]) const;

Purpose: Gets the user's address

Arguments: Three char arrays and two integers making up the user's address

Return Value: Void

void SetFirstName(const char* afName);

Purpose: Sets the user's first name

Arguments: A char array of the user's new first name

Return Value: Void

void SetLastName(const char* alName);

Purpose: Sets the user's last name

Arguments: A char array of the user's new last name

Return Value: Void

void SetMajor(const char* amajor);

Purpose: Sets the user's major

Arguments: A char array of the user's new major

Return Value: Void

void SetEmail(const char* anemail);

Purpose: Sets the user's email

Arguments: A char array of the user's new email

Return Value: Void

void SetGender(char aGender);

Purpose: Sets the user's gender

Arguments: A char representing the user's gender

Return Value: Void

void SetDateOfBirth(DateType aDateOfBirth);

Purpose: Sets the user's date of birth

Arguments: A DateType of three integers representing the user's new birthday

Return Value: Void

void SetGPA(float aGPA);

Purpose: Sets the user's GPA

Arguments: A float representing the user's new GPA

Return Value: Void

void SetAddress(AddressType aAddress);

Purpose: Sets the user's address

Arguments: An AddressType of three char arrays and two integers representing the user's new address

Return Value: Void

void SetAddress(char aStreetName[], int aStreetNo, char aCity[], int aZip, char aState[]);

Purpose: Sets the user's new address

Arguments: Three char arrays and two integers representing the user's new address

Return Value: Void

void Display() const;

Purpose: Displays all the data from the user to the console

Arguments: None

Return Value: Void

void Display(std::ofstream& outFile) const;

Purpose: Prints all the data of the user into a file

Arguments: An ofstream representing the file being printed to

Return Value: Void

Class: Test Driver

int Populate(const char* input, User users[]);

Purpose: Reads in all users from a file into an users array

Arguments: A char array representing the file to read from and a users array to store the data into

Return Value: An integer representing the number of users that were read in

void Test(User users[], int count) const;

Purpose: Tests the user class and its helpers class, printing results to the console

Arguments: An users array to test on and an integer representing the number of users in the user array.

Return Value: Void

void Test(std::ofstream& outFile, User users[], int count) const;

Purpose: Tests the user class and its helpers class, printing results to a file

Arguments: An ofstream representing the file being printed to, an users array to test on and an integer representing the number of users in the user array.

Return Value: Void

Test Plan

This software is tested by the exhaustive method. Every method is called and every variable is used to confirm functionality.

Section II.1. Test Procedures

Main Testing

•Purpose - To test most of the methods

•Procedure - Reads in users and then displays them

•Inputs - Users

•Expected Outputs - The same Users

•Success Criteria - Inputs match the outputs

Secondary Testing

•Purpose - To test the remaining redundant methods

•Procedure - Alters current users and then displays alterations

•Inputs - Users

•Expected Outputs - The altered Users

•Success Criteria - Outputs are successfully altered and do NOT match the inputs

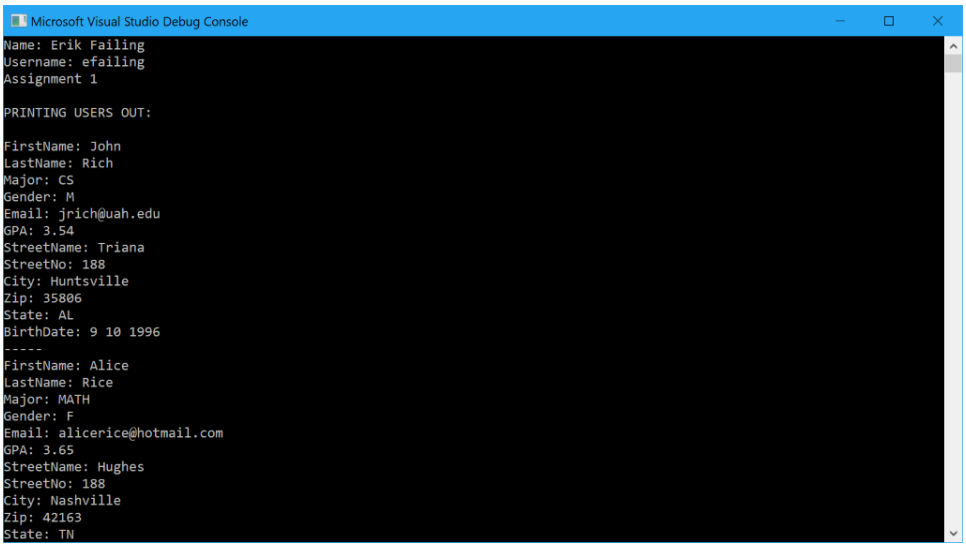
Section II.2. Sample Runs

Sample Run 1:

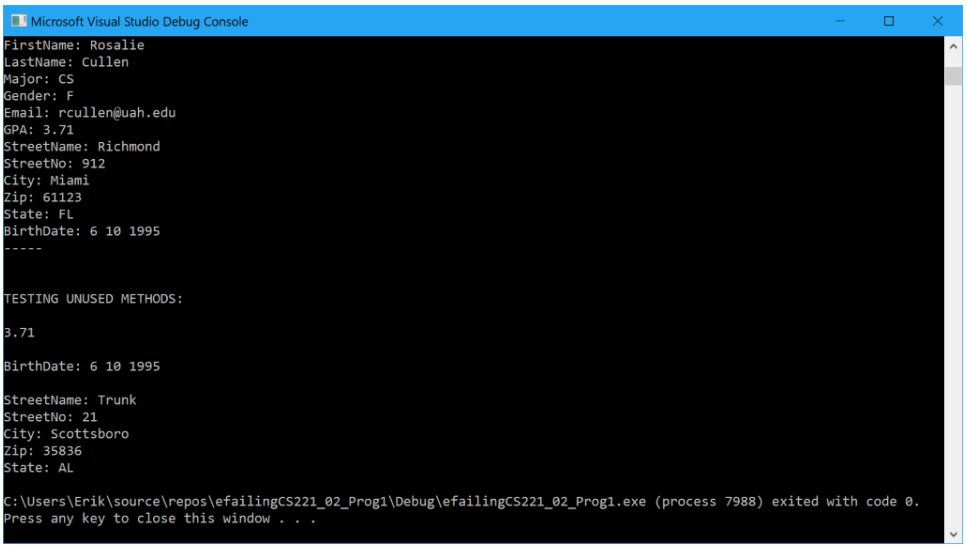
•Inputs - Sample user text file hw1samplefile.txt

•Outputs - efailingHw1Outfile.txt

•Snapshots:


Microsoft Visual Studio Debug Console
Name: Erik Failing
Username: efailing
Assignment 1
PRINTING USERS OUT:
FirstName: John
LastName: Rich
Major: CS
Gender: M
Email: jrich@uah.edu
GPA: 3.54
StreetName: Triana
StreetNo: 188
City: Huntsville
Zip: 35806
State: AL
BirthDate: 9 10 1996

FirstName: Alice
LastName: Rice
Major: MATH
Gender: F
Email: alicerice@hotmail.com
GPA: 3.65
StreetName: Hughes
StreetNo: 188
City: Nashville
Zip: 42163
State: TN


Microsoft Visual Studio Debug Console
FirstName: Rosalie
LastName: Cullen
Major: CS
Gender: F
Email: rcullen@uah.edu
GPA: 3.71
StreetName: Richmond
StreetNo: 912
City: Miami
Zip: 61123
State: FL
BirthDate: 6 10 1995

TESTING UNUSED METHODS:
3.71
BirthDate: 6 10 1995
StreetName: Trunk
StreetNo: 21
City: Scottsboro
Zip: 35836
State: AL
C:\Users\Erik\source\repos\efailingCS221_02_Prog1\Debug\efailingCS221_02_Prog1.exe (process 7988) exited with code 0.
Press any key to close this window . . .