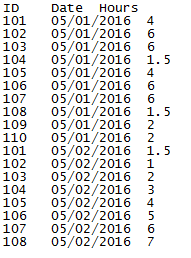
Sarah Wood

Agile Programming

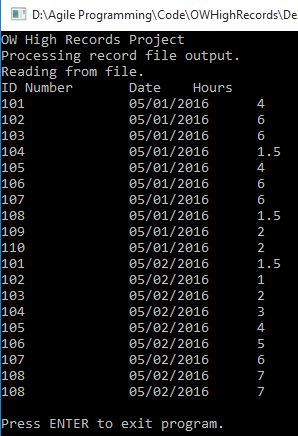
P2-Deliverable 1

It took me two hours to write this. I spent about an hour writing class constructors & member functions before I realized the spec was using Java-esque language. I then wrote a FUNCTION for file I/O. The “class” work will be useful later.

Output File



Screen Cap



Source Code (relevant parts)

&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&

StudentRecords.h

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

#ifndef RECORDS\_H

#define RECORDS\_H

#include <iostream>

#include <string>

#include <fstream>

#include <ctime>

using namespace std; //no custom namespace is required

//records structure declarations

struct StudentDailyAttendance

{

//Student has an array of days with attendance

int IDNumber;

string Date;

int ClassRoomNumber;

int ClockINTimeCode;

int ClockOUTTimeCode;

double TimeInClass;

StudentDailyAttendance()

{

IDNumber = 000;

ClassRoomNumber = 999;

Date = "0/0/0";

ClockINTimeCode = 0000;

ClockOUTTimeCode = 0000;

TimeInClass = 0.0;

}

};

#endif

&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&

Source.cpp

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

/\*

Sarah Wood

Agile Programming

OW High Records Project

May 2016

Template for file output:

Student ID Date Time

3141592 1/2/1234 4.00

1234586 2/1/4321 2.50

Cpp files I will want to review:

ch16, a 15 & 16

ch15, a 3, 12, 13

Final Project

\*/

#include <iostream>

#include<iomanip>

#include <string>

#include <fstream>

#include <ctime>

#include "Person.h"

#include "StudentRecords.h"

using namespace std; //no custom namespace is required

const int NUM\_OF\_DAYS = 367; //2016 is a leap year, 0 holds no data

const int NUM\_OF\_STUDENTS = 11;

//make an array of generic students

//Student(int ID, string name, string role, int grade, string cont1, string cont2, string cont3, string ed1, string ed2, string ed3)

Student ArrayOfStudents[NUM\_OF\_STUDENTS] = {

Student(100, "Test Entry", "Student", 10, "111-111-1111", "", "", "Test Elementary","",""),

Student(101, "Bart Simpson", "Student", 10, "Mom Marge 222-222-2222", " ", " " , "Springfield Elementary", "Springfield Middle School", "Grade 4: Springfield School for the Gifted"),

Student(102, "Lisa Simpson", "Student", 10, "Mom Marge 222-222-2222", "", "", "Springfield Elementary", "Springfield Middle School", ""),

Student(103, "Morty Smith", "Student", 10, "Mom Beth 333-333-3333", "Dad Jerry 444-444-4444", "Grandpa Rick 555-555-5555", "", "", "" ),

Student(104, "Summer Smith", "Student", 10, "Mom Beth 333-333-3333", "Dad Jerry 444-444-4444", "Grandpa Rick 555-555-5555", "", "", ""),

Student(105, "Eric Cartman", "Student", 10, "Mom Liane 777-777-7777","","","South Park Elementary", "",""),

Student(106, "Stan Marsh", "Student", 10, "Dad Randy 888-888-8888", "Mom Sharon 999-999-9999", "", "South Park Elementary", "", ""),

Student(107, "Kyle Broflovski", "Student", 10, "Mom Sheila 123-456-7890", "Dad Gerald 234-567-8901", "", "South Park Elementary", "", ""),

Student(108, "Kenny McCormick", "Student", 10, "Mom Carol 345-678-9012", "Dad Stuart 345-678-9012", "", "South Park Elementary", "", ""),

Student(109, "Beavis Cornholio", "Student", 10, "Mom 456-789-0123", "", "", "","","" ),

Student(110, "Butt Head", "Student", 10, "Mom 567-890-1234","","","","",""),

};

//Build an array of attendance records for each student

StudentDailyAttendance AttendanceArray[9999] = {};

//function declarations

int fileIOforStudentRecords(Student\*, StudentDailyAttendance\*);

void main()

{

//init console

cout << "OW High Records Project" << endl;

fileIOforStudentRecords(ArrayOfStudents, AttendanceArray);

//keep window open until we want to close it

cout << '\n' << "Press ENTER to exit program.";

cin.ignore(256, '\n');//ignore up to 256 characters, stop ignoring at any instance of '\n'

cin.get();

}

int fileIOforStudentRecords(Student \*arr ,StudentDailyAttendance \*att)

{

fstream recordOUT, recordIN;

string tempDate;

int tempID;

double tempHrs;

cout << "Processing record file output." << endl;

//open file, write

recordOUT.open("testrecords2.txt");

if (recordOUT)

{

recordOUT << "ID Number" << '\t' << "Date" << '\t' << "Hours" << '\n';

for (int i = 1; i < 20; i++)

{

recordOUT << att[i].IDNumber << '\t' << att[i].Date << '\t' << att[i].TimeInClass << '\n';

}

}

else

{

cout << "Record File Output Error." << endl;

return 0;

}

//open file, read

recordIN.open("testrecord.txt");

if (recordIN)

{

cout << "Reading from file." << '\n';

cout << "ID Number" << '\t' << "Date" << '\t' << "Hours" << '\n';

recordIN.ignore(256, '\n'); //ignore the first line: it's just headers

for (int i=1; !recordIN.eof(); i++) // "i" will correspond to the unique attendance entry ID.

{

recordIN >> tempID >> tempDate >> tempHrs;

att[i].IDNumber = tempID;

att[i].Date = tempDate;

att[i].TimeInClass = tempHrs;

cout << setprecision(2);

cout <<tempID << '\t' << '\t' << tempDate << '\t' << tempHrs << '\n';

i++;

}

}

else

{

cout << "Record File Input Error." << endl;

return 0;

}

return 1;

}