PROJECT 1

GAME SCORE

CSCI 220
DATA STRUCTURE 1

MAI PHAM

DEVELOPMENT ENVIRONMENT
MacOS – Xcode
Window 10 – MSVS 2017

TABLE OF CONTENTS

Mai Pham
CSCI 220 - Project 1
Prof. T. Vo

PROJECT NOTE

OBJECTIVE:

- create two classes; one class to hold a game entry and the other class game score to hold a list of 10 game entries.

SUMMARY:

- This project helps me understand the 'has-a' relationship in OOP, which I'm struggling on. Besides that, I have a couples 'off by one' errors in my GameScore class which caused the program to product wrong information.

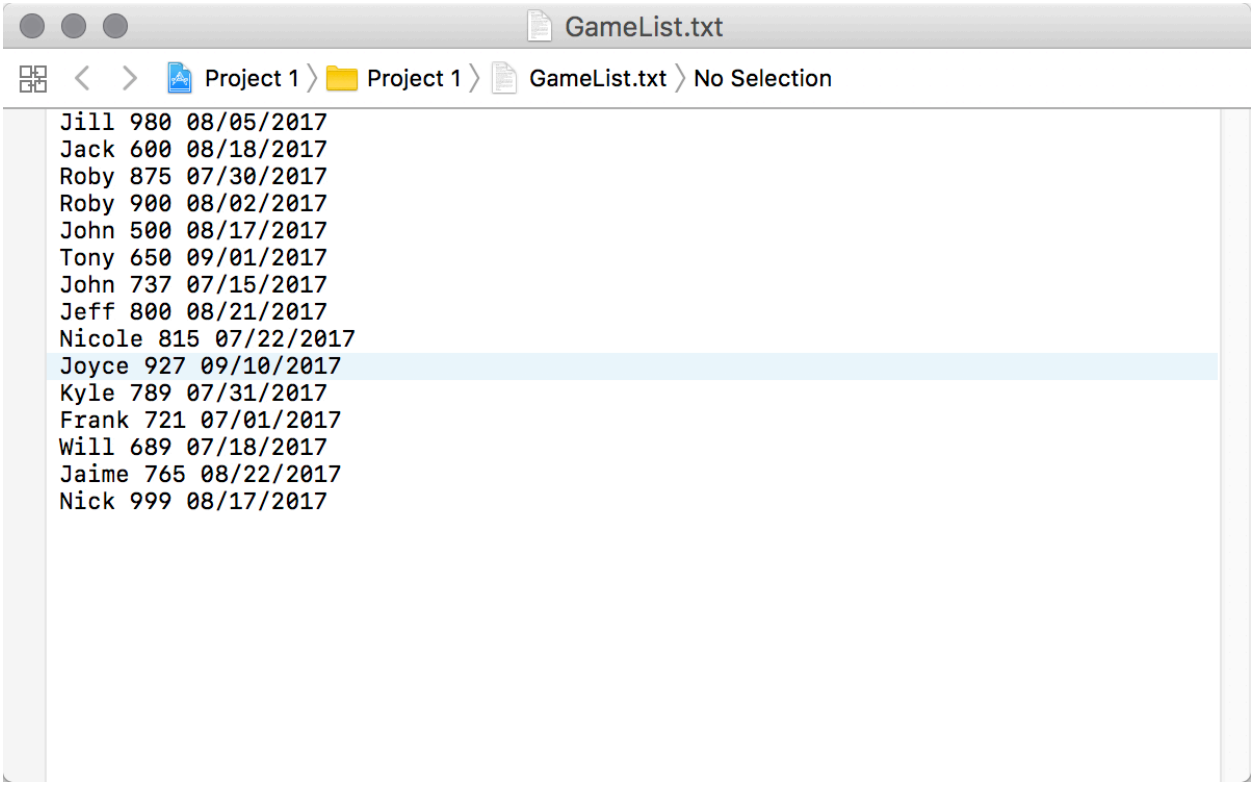
EXTRA CREDIT:

- I did both extra credit 1 and 2. Extra credit 2 is not hard. However, I did spend a lot of time on extra credit 1. While working on extra credit 1, my program didn't output any information. I thought it is due to the input/output file error since I just recently work on Xcode. However, it turns out that my coding has some syntax and logic errors that the Xcode didn't complain about. So, I have to go back to Microsoft Visual Studio to do the debugging before continues working on Xcode again.

CONCLUSION:

- Overall, my project is completed and successfully run the main part and 2 extra credits.

INPUT FILE



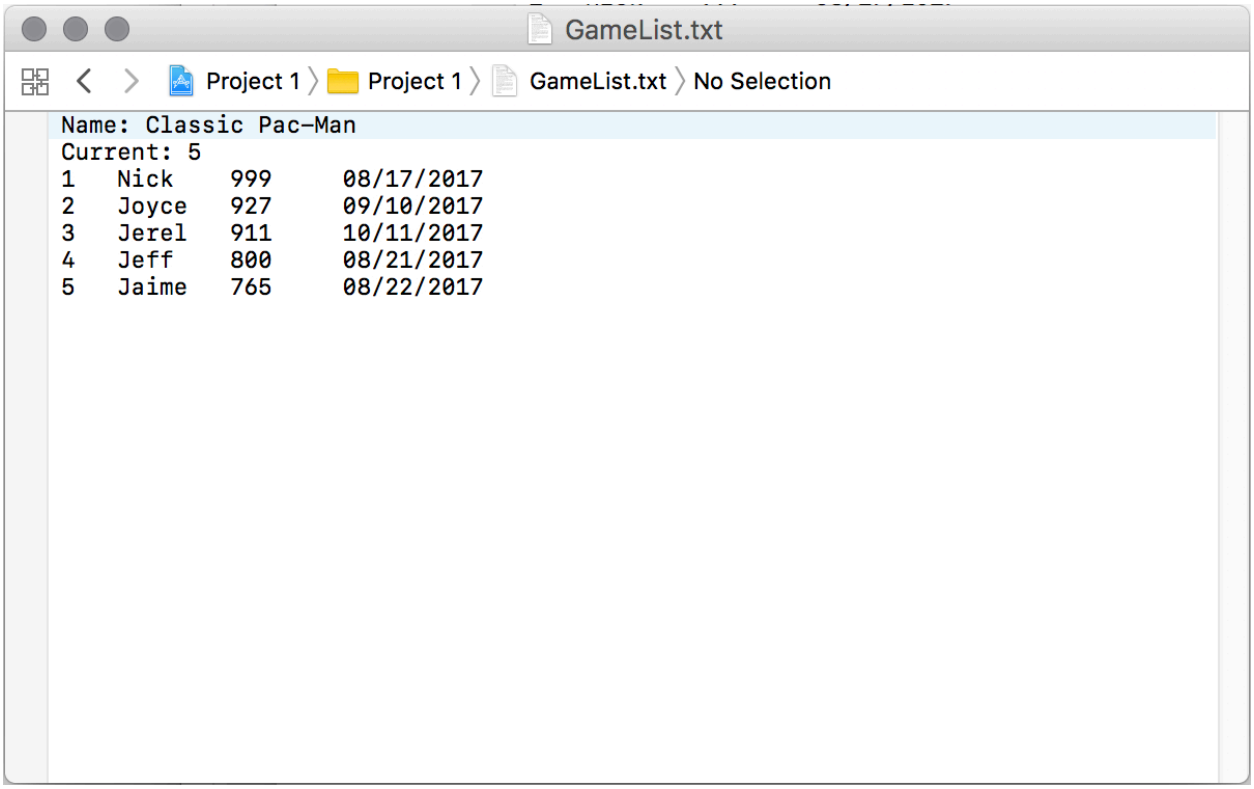
OUTPUT FILE

OUTPUT WINDOW

```
Project 1 for DATA STRUCTURE 1 (PROF. T.VO)
Author: Mai Pham

Display input file data.
Jill    980     08/05/2017
Jack    600     08/18/2017
Roby    875     07/30/2017
Roby    900     08/02/2017
John    500     08/17/2017
Tony    650     09/01/2017
John    737     07/15/2017
Jeff    800     08/21/2017
Nicole  815     07/22/2017
Joyce   927     09/10/2017
Kyle    789     07/31/2017
Frank   721     07/01/2017
Will    689     07/18/2017
Jaime   765     08/22/2017
Nick    999     08/17/2017

Display Game Score List.
Name: Classic Pac-Man
Current: 10
1   Nick    999     08/17/2017
2   Jill    980     08/05/2017
3   Joyce   927     09/10/2017
4   Roby    900     08/02/2017
5   Roby    875     07/30/2017
6   Nicole  815     07/22/2017
7   Jeff    800     08/21/2017
8   Kyle    789     07/31/2017
9   Jaime   765     08/22/2017
10  John    737     07/15/2017

Dislay entry #5:
Entry #5: Roby 875 07/30/2017

Remove entry #5 and display new list.
Name: Classic Pac-Man
Current: 9
1   Nick    999     08/17/2017
2   Jill    980     08/05/2017
3   Joyce   927     09/10/2017
4   Roby    900     08/02/2017
5   Nicole  815     07/22/2017
6   Jeff    800     08/21/2017
7   Kyle    789     07/31/2017
8   Jaime   765     08/22/2017
9   John    737     07/15/2017
```

```
Remove all game entry before 08/15/2017.
Name: Classic Pac-Man
Current: 4
1   Nick    999     08/17/2017
2   Joyce   927     09/10/2017
3   Jeff    800     08/21/2017
4   Jaime   765     08/22/2017

Add one entry, display, and save the new list into file.
Name: Classic Pac-Man
Current: 5
1   Nick    999     08/17/2017
2   Joyce   927     09/10/2017
3   Jerel   911     10/11/2017
4   Jeff    800     08/21/2017
5   Jaime   765     08/22/2017

Program ended with exit code: 0
```

TEXT FILE



SOURCE CODE

GAME ENTRY

Header File

```
//
//  GameEntry.h
//  Project 1 for DATA STRUCTURE 1 (PROF. T.VO)
```

```
//
//  Created by Mai Pham on 9/5/17.
//  Copyright © 2017 Mai Pham. All rights reserved.
//

#ifndef GameEntry_h
#define GameEntry_h

#include <iostream>
#include <string>
#include <fstream>
using namespace std;

class GameEntry
{
private:
    string playerName, date;
    int score;

public:
    GameEntry();
    GameEntry(string name);
    GameEntry(string name, int point,  string d);
    void setName(string name);
    void setScore(int point);
    void setDate(string d);
    string getName();
    int getScore();
    string getDate();
};

#endif
```

Implementation File

```
//
//  GameEntry.cpp
//  Project 1 for DATA STRUCTURE 1 (PROF. T.VO)
//
//  Created by Mai Pham on 9/5/17.
//  Copyright © 2017 Mai Pham. All rights reserved.
//
//  Implementation File (Member Functions)
//

#include "GameEntry.h"

#include <iostream>
#include <string>
#include <fstream>
using namespace std;

GameEntry::GameEntry()
{
    playerName = "";
    score = 0;
    date = "00/00/00";
}
GameEntry::GameEntry(string name)
{
    playerName = name;
}
GameEntry::GameEntry(string name, int point, string d)
{
    playerName = name;
```

```
    score = point;
    date = d;
}
void GameEntry::setName(string name)
{
    playerName = name;
}
void GameEntry::setScore(int point)
{
    score = point;
}
void GameEntry::setDate(string d)
{
    date = d;
}
string GameEntry::getName()
{
    return playerName;
}
int GameEntry::getScore()
{
    return score;
}
string GameEntry::getDate()
{
    return date;
}
```

GAME SCORE
Header File

```
//
//  GameScore.h
//  Project 1 for DATA STRUCTURE 1 (PROF. T.VO)
//
//  Created by Mai Pham on 9/5/17.
//  Copyright © 2017 Mai Pham. All rights reserved.
//

#ifndef GameScore_h
#define GameScore_h

#include "GameEntry.h"

#include <iostream>
#include <string>
#include <fstream>
using namespace std;

const int MAX_ENTRIES = 10;

class GameScore
{
private:
    GameEntry list[MAX_ENTRIES];
    int currentEntries;
    string gameName;

public:
    GameScore(string name);
    void getEntry(int n);
    void add(GameEntry e);
    void remove(int n);
    void remove(string d);
    void print();
```

```
    void print(string name);
};

#endif
```

Implementation File

```
//
//  GameScore.cpp
//  Project 1 for DATA STRUCTURE 1 (PROF. T.VO)
//
//  Created by Mai Pham on 9/5/17.
//  Copyright © 2017 Mai Pham. All rights reserved.
//
//  Implementation File (Member Functions)
//

#include "GameScore.h"
#include "GameEntry.h"

#include <iostream>
#include <string>
#include <fstream>
using namespace std;

GameScore::GameScore(string name)
{
    gameName = name;
    currentEntries = 0;
}
void GameScore::getEntry(int n)
{
    cout << "Entry #" << n << ": ";
    n--;
    cout << list[n].getName() << " " << list[n].getScore() << " " << list[n].getDate() <<
endl;
}
void GameScore::add(GameEntry e)
{
    GameEntry temp;
    int count = currentEntries;
    if (currentEntries == MAX_ENTRIES)
    {
        if(e.getScore() > list[MAX_ENTRIES-1].getScore())
        {
            for (int i = currentEntries-1; i >= 0; i--)
            {
                if (list[i].getScore() < e.getScore())
                {
                    list[i] = list[i-1];
                    count--;
                }
            }
            list[count] = e;
        }
    }
    else
    {
        list[currentEntries] = e;
        currentEntries++;
        for (int i = 0; i < currentEntries; i++)
        {
            for (int j = 0; j < (currentEntries - 1); j++)
            {
                if (list[j].getScore() < list[j+1].getScore())
                {
```

```
                    temp = list[j];
                    list[j] = list[j+1];
                    list[j+1] = temp;
                }
            }
        }
    }
}
void GameScore::remove(int n)
{
    n--;
    for (int i = n; i < currentEntries-1; i++)
        list[i] = list[i+1];
    currentEntries--;
}
void GameScore::remove(string d)
{
    int i = 0;
    while (i < currentEntries)
    {
        if(list[i].getDate() < d)
            remove(i+1);
        else
            i++;
    }
}
void GameScore::print()
{
    cout << "Name: " << gameName << endl;
    cout << "Current: " << currentEntries << endl;
    for (int i = 0; i < currentEntries; i++)
        cout << (i+1) << "\t" << list[i].getName() << "\t"
        << list[i].getScore() << "\t\t" << list[i].getDate() << endl;
    cout << endl;
}
void GameScore::print(string fileName)
{
    ofstream outFile;
    outFile.open(fileName);

    outFile << "Name: " << gameName << endl;
    outFile << "Current: " << currentEntries << endl;
    for (int i = 0; i < currentEntries; i++)
        outFile << (i+1) << "\t" << list[i].getName() << "\t"
        << list[i].getScore() << "\t\t" << list[i].getDate() << endl;
    cout << endl;
    outFile.close();
}
```

MAIN

```
//
//  main.cpp
//  Project 1 for DATA STRUCTURE 1 (PROF. T.VO)
//
//  Created by Mai Pham on 9/5/17.
//  Copyright © 2017 Mai Pham. All rights reserved.
//

#include "GameScore.h"
#include "GameEntry.h"

#include <iostream>
#include <string>
#include <fstream>
```

```
using namespace std;

int main()
{
    string name, date;
    int score;

    ifstream inputFile;
    inputFile.open("GameList.txt");

    GameScore g1("Classic Pac-Man");

    if (!inputFile)
    {
        cout << "Error opening file. \n";
        cout << "The file was not found" << endl;
        return 1;
    }

    cout << "Project 1 for DATA STRUCTURE 1 (PROF. T.VO)\n";
    cout << "Author: Mai Pham\n\n";

    cout << "Display input file data.\n";
    inputFile >> name;
    while (!inputFile.eof())
    {
        inputFile >> score >> date;
        cout << name << "\t" << score << "\t\t" << date << endl;

        GameEntry gamePerson (name, score, date);
        g1.add(gamePerson);

        inputFile >> name;
    }
    cout << "\nDisplay Game Score List.\n";
    g1.print();

    cout << "Dislay entry #5:\n";
    g1.getEntry(5);

    cout << "\nRemove entry #5 and display new list.\n";
    g1.remove(5);
    g1.print();

    cout << "Remove all game entry before 08/15/2017.\n";
    g1.remove("08/15/2017");
    g1.print();

    cout << "Add one entry, display, and save the new list into file.";
    GameEntry gamePerson ("Jerel", 911, "10/11/2017");
    g1.add(gamePerson);
    g1.print("GameList.txt");
    g1.print();

    inputFile.close();

    return 0;
}
```