

screen.io

```
1 *****
2 * PROGRAMMED BY : Negin Mashhadi
3 * STUDENT ID    : 1084104
4 * CLASS        : CS1B - MW - 6:30pm
5 * ASSIGNMENT #1 : Functions and Arrays
6 *****
7
8
9 What input file would you like to use? inFile.txt
10 What output file would you like to use? outFile.txt
11
12
13 MENU OPTIONS
14
15 1 - Find the larger balance
16 2 - Find the smaller balance
17 3 - Obtain the sum of all balances
18 4 - Obtain the average of all balances
19 5 - Find person
20 0 - Exit
21 Enter an option (0 to exit): 1
22
23 Finding the larger balance...
24
25
26 MENU OPTIONS
27
28 1 - Find the larger balance
29 2 - Find the smaller balance
30 3 - Obtain the sum of all balances
31 4 - Obtain the average of all balances
32 5 - Find person
33 0 - Exit
34 Enter an option (0 to exit): 2
35
36 Finding the smaller balance...
37
38
39 MENU OPTIONS
40
41 1 - Find the larger balance
42 2 - Find the smaller balance
43 3 - Obtain the sum of all balances
44 4 - Obtain the average of all balances
45 5 - Find person
46 0 - Exit
47 Enter an option (0 to exit): 3
48
49 Obtaining the sum of all balances...
50
51
52 MENU OPTIONS
53
54 1 - Find the larger balance
55 2 - Find the smaller balance
56 3 - Obtain the sum of all balances
57 4 - Obtain the average of all balances
```

```
585 - Find person
590 - Exit
60Enter an option (0 to exit): 4
61
62Obtaining the average of all balances...
63
64
65MENU OPTIONS
66
671 - Find the larger balance
682 - Find the smaller balance
693 - Obtain the sum of all balances
704 - Obtain the average of all balances
715 - Find person
720 - Exit
73Enter an option (0 to exit): 5
74
75Who would you like to search for: Steve Woolston
76Found.
77
78
79
80MENU OPTIONS
81
821 - Find the larger balance
832 - Find the smaller balance
843 - Obtain the sum of all balances
854 - Obtain the average of all balances
865 - Find person
870 - Exit
88Enter an option (0 to exit): 5
89
90Who would you like to search for: Jacques Rousseau
91Jacques Rousseau was not found.
92
93
94MENU OPTIONS
95
961 - Find the larger balance
972 - Find the smaller balance
983 - Obtain the sum of all balances
994 - Obtain the average of all balances
1005 - Find person
1010 - Exit
102Enter an option (0 to exit): 5
103
104Who would you like to search for: Chris Carroll
105Found.
106
107
108
109MENU OPTIONS
110
1111 - Find the larger balance
1122 - Find the smaller balance
1133 - Obtain the sum of all balances
1144 - Obtain the average of all balances
```

screen.io

```
1155 - Find person
1160 - Exit
117Enter an option (0 to exit): 5
118
119Who would you like to search for: Pete McBride
120Found.
121
122
123
124MENU OPTIONS
125
1261 - Find the larger balance
1272 - Find the smaller balance
1283 - Obtain the sum of all balances
1294 - Obtain the average of all balances
1305 - Find person
1310 - Exit
132Enter an option (0 to exit): 5
133
134Who would you like to search for: Jean Rousseau
135Found.
136
137
138
139MENU OPTIONS
140
1411 - Find the larger balance
1422 - Find the smaller balance
1433 - Obtain the sum of all balances
1444 - Obtain the average of all balances
1455 - Find person
1460 - Exit
147Enter an option (0 to exit): 5
148
149Who would you like to search for: Florance Cyr
150Florance Cyr was not found.
151
152
153MENU OPTIONS
154
1551 - Find the larger balance
1562 - Find the smaller balance
1573 - Obtain the sum of all balances
1584 - Obtain the average of all balances
1595 - Find person
1600 - Exit
161Enter an option (0 to exit): 6
162
163
164Thank you for using my program.
```

```

1 *****
2 * PROGRAMMED BY : Negin Mashhadi
3 * STUDENT ID    : 1084104
4 * CLASS        : CS1B - MW - 6:30pm
5 * ASSIGNMENT #1 : Functions and Arrays
6 *****
7
8
9 Largest Balance:
10 ID #    NAME                                BALANCE DUE
11 ----    -
12 1002    Steve Woolston                      $    1423.2
13
14 Smallest Balance:
15 ID #    NAME                                BALANCE DUE
16 ----    -
17 1003    Don McBride                          $     12.32
18
19 Sum of balance for all persons:
20 $    4080.48
21
22 Average of balance for all persons:
23 $     408.05
24
25
26 Search Name:
27 ID #    NAME                                BALANCE DUE
28 ----    -
29 1002    Steve Woolston                      $    1423.20
30
31 Search Name:
32 ID #    NAME                                BALANCE DUE
33 ----    -
34 1008    Chris Carroll                       $     32.35
35
36 Search Name:
37 ID #    NAME                                BALANCE DUE
38 ----    -
39 1007    Pete McBride                        $    500.32
40
41 Search Name:
42 ID #    NAME                                BALANCE DUE
43 ----    -
44 1001    Jean Rousseau                       $     15.50
45

```

# assignment1HeaderFile.h

```

1 #ifndef ASSIGNMENT1HEADERFILE_H_
2 #define ASSIGNMENT1HEADERFILE_H_
3
4 //processor directive go here
5 #include <iostream>
6 #include <iomanip>      /**setw**/
7 #include <string>       /**Strings**/
8 #include <fstream>      /**fout**/
9 #include <ostream>      /**cout or fout**/
10 using namespace std;
11
12
13 //GLOBAL CONSTANTS
14 const int AR_SIZE_STRING = 6;
15 const string MENU = {"\n\nMENU OPTIONS\n\n"
16                     "1 - Find the larger balance\n"
17                     "2 - Find the smaller balance\n"
18                     "3 - Obtain the sum of all balances\n"
19                     "4 - Obtain the average of all "
20                     "balances\n"
21                     "5 - Find person\n"
22                     "0 - Exit\n"
23                     "Enter an option (0 to exit): "
24                     };
25
26 const string MENU_LIST[AR_SIZE_STRING] =
27 {"\nThank you for using my program.",
28 "\nFinding the larger balance...",
29 "\nFinding the smaller balance...",
30 "\nObtaining the sum of all balances...",
31 "\nObtaining the average of all balances...",
32 "\nWho would you like to search for: "
33 };
34 enum menuOption
35 {
36     EXIT,          //0
37     LARGEST,        //1
38     SMALLEST,      //2
39     SUM,           //3
40     AVG,           //4
41     FINDPERSON     //5
42 };
43
44 //PROTOTYPES
45
46
47 /*****
48 * PrintHeader
49 * This function receives an assignment name, type and number then outputs
50 * the appropriate header
51 * ==> returns nothing - This will output the class heading.
52 *****/
53 void PrintHeader (ostream &output, // IN/OUT - output file
54                  string asName, //IN - assignment Name - used for output
55                  char asType, //IN - assignment Type
56                  // - (LAB or ASSIGN) - used for output
57                  int asNum); //IN - assignment Name - used for output

```

# assignment1HeaderFile.h

```

58
59
60 /*****
61 * InputFunction
62 *   This function will receive an input file and read 3 sets of data from
63 * the file. It will create three parallel arrays name, Id and the balance.
64 *   * ==> returns nothing - This will read data from input file.
65 *****/
66 void InputFunction ( string finFileName, // IN - input file name
67                     string fnameAr[],   // IN - read in strings from file to the
68                                     // array
69                     int fidsAr[],        // IN - read in ints from file to the
70                                     // array
71                     double fbalanceAr[], // IN - read in ints from file to the
72                                     // array
73                     const int AR_SIZE); // IN - the size of the array
74
75 /*****
76 * SearchBalance Function
77 *   This function will search for the largest or smallest balance in the
78 * array of balances based on the user input. it will then return the index
79 * of the desired balance of the user.
80 *   * ==> returns the index of the largest or smallest number
81 *****/
82 int SearchBalance (int fOption, // IN - Option input of the user
83                   double fBalanceAr[], // IN - The balance array read from file
84                   const int AR_SIZE); // IN - The array size
85
86 /*****
87 * SumAndAverage Function
88 *   This function will calculate the sum and average of all the balances in
89 * balance array based on the option the user chooses
90 *   * ==> returns either the sum of all the balances or average
91 *****/
92 double SumAndAverage (int fOption, // IN - Option input of the user
93                       double fBalanceAr[], // IN - The balance array read from
94                                       // file
95                       const int AR_SIZE); // IN - The array size
96
97 /*****
98 * SearchName
99 *   This function will prompt the user to enter the name of the person they
100 * are looking for. The function will search search for the name through the
101 * name array.
102 *   * ==> returns the index of the name in the nameAr.
103 *****/
104 int SearchName( string fNameAr[], // IN - The name array
105                const int AR_SIZE); // IN - The array size
106 /*****
107 * PrintOptions
108 *   This function Will output the results of users search in to the output
109 * file. It will specifically output the results for find largest balance,
110 * smallest balance, and person.
111 *   ==> returns nothing - This will output the results.
112 *****/
113 void PrintOptions(ofstream &fout, // IN/OUT - output file
114                  int fIdsAr[], // IN/CALC - the ids array

```

# assignment1HeaderFile.h

```
115         string fNameAr[],      // IN/CALC - The name array
116         double fBalanceAr[],    // IN/CALC - The balance array
117         int     index,          // CALC    - The index of the array
118         int     fOption);       // IN      - Users choice of the menu
119 /*****
120 *   PrintOptions
121 *       This function Will output the results of users search in to the output
122 *   file. It will specifically output the results for sum and average.
123 *       ==> returns nothing - This will output the results.
124 *****/
125 void PrintSumAndAvg(ofstream &fout,      // IN/OUT - output file
126                    double     fBalance,  // CALC    - The sum or average
127                    const string prompt); // OUT     - The prompt for output
128
129
130 #endif /* ASSIGNMENT1HEADERFILE_H_ */
131
```

## main.cpp

```
1/*****
2 * AURHOR      : Negin Mashhadi
3 * STUDENT ID   : 1084104
4 * ASSIGNMENT#1 : Functions and Arrays
5 * CLASS        : CS1B
6 * SECTION      : MW - 6:30pm - 9:50pm
7 * DUE DATE     : 1/29/2018
8 * *****/
9#include "assignment1HeaderFile.h"
10/*****
11 * Functions and Arrays
12 * -----
13 * This program will ask the user to input the name of a input file name and
14 * an output file. It will read the in a list of names, id #'s, and balances
15 * from the specified input file and will initialize a three parallel array.
16 * A menu will be provided for the user which allows the user to choose for
17 * specific executions to happen.
18 * -----
19 * INPUT
20 *      inFileName : The name of the input file
21 *      outFileNam : The name of the output file
22 *      options    : The number of the menu option the user will enter
23 * OUTPUT
24 *
25 *****/
26int main()
27{
28/*****
29 * CONSTANTS
30 * -----
31 * PROCESSING - the following is used for the size of the arrays used in this
32 * program
33 * -----
34 * AR_SIZE      : The size of the array
35 * COL_SIZE_PROMPT : The column size for the prompt
36 *****/
37     const int PROMPT_COL_SIZE = 40;
38     const int AR_SIZE         = 10;
39
40
41     const string SUM_PROMPT = "\nSum of balance for all persons: \n";
42     const string AVG_PROMPT = "Average of balance for all persons: \n";
43
44     //VARIABLE DECLARATION
45     ifstream inFile;
46     ofstream outFile;
47
48     string nameAr[AR_SIZE];    // IN   - Name array
49     int idsAr[AR_SIZE];        // IN   - ids array
50     double balanceAr[AR_SIZE]; // IN   - balances array
51
52     string inFileNam;          // IN   - Input file name entered by user
53     string outFileNam;         // IN   - Output file name entered by user
54     int option;                // IN   - User input based on what they want
55                                //      from the menu
56     int balanceValueLargest;    // CALC - The index of the largest balance
57     int balanceValueSmallest;  // CALC - The index of the smallest balance
```



main.cpp

```
58     double sum;                // CALC - Holds the sum of balances
59     double avg;                // CALC - Holds the avg of balance
60     string searchName;         // IN   - The name being searched
61     int searchPerson ;         // CALC - Holds the index for name searched
62     bool valid;                // CALC - Checks for valid inputs
63     menuOption optionMenu;     // CALC - The enum of the option user enters
64
65
66
67     /*INTIALIZINH*/
68     balanceValueLargest      =   -1;
69     balanceValueSmallest    =   -1;
70
71     /*****
72     * INPUT - The name of the input file and output file
73     *****/
74     PrintHeader( cout, "Functions and Arrays", 'A' , 1);
75     cout << left << setw(PROMPT_COL_SIZE)
76         << "What input file would you like to use? ";
77         getline(cin, inFileName);
78
79     cout << "What output file would you like to use? ";
80     getline(cin, outFileName);
81
82     /*****
83     * PROCESSING - performing a certain task based on user input
84     *****/
85     oFile.open(outFileName);
86     PrintHeader( oFile, "Functions and Arrays", 'A' , 1);
87
88
89     InputFunctin(inFileName, nameAr, idsAr, balanceAr, AR_SIZE);
90
91
92     do{
93         cout << MENU;
94         cin >> option;
95         cin.ignore(1000, '\n');
96
97         valid = (option == 0)||
98                 (option == 1)||
99                 (option == 2)||
100                (option == 3)||
101                (option == 4)||
102                (option == 5);
103
104
105         if (valid)
106         {
107             optionMenu = menuOption(option);
108             cout << MENU_LIST[optionMenu];
109         }
110         cout << endl;
111
112
113         switch(option)
114         {
```

main.cpp

```
115         case LARGEST :
116             balanceValueLargest = SearchBalance(option, balanceAr,
117                                                 AR_SIZE);
118             PrintOptions(oFile, idsAr, nameAr ,balanceAr,
119                         balanceValueLargest, option);
120             break;
121
122         case SMALLEST:
123             balanceValueSmallest =
124                 SearchBalance(option, balanceAr, AR_SIZE);
125
126             PrintOptions(oFile, idsAr, nameAr ,balanceAr,
127                         balanceValueSmallest, option);
128             break;
129         case SUM:
130             sum = SumAndAverage(option, balanceAr, AR_SIZE);
131             PrintSumAndAvg(oFile, sum, SUM_PROMPT);
132             break;
133         case AVG:
134             avg = SumAndAverage(option, balanceAr, AR_SIZE);
135             PrintSumAndAvg(oFile, avg, AVG_PROMPT);
136             break;
137         case FINDPERSON:
138             searchPerson = SearchName(nameAr, AR_SIZE);
139             PrintOptions(oFile, idsAr, nameAr ,
140                         balanceAr, searchPerson, option);
141             break;
142     }
143
144     if(!valid)
145     {
146         cout << MENU_LIST[EXIT];
147     }
148
149 }while(valid && option != 0);
150
151 oFile.close();
152
153 return 0;
154 }
155
```

## InputFunction.cpp

```

1 /*****
2  * AURHOR      :   Negin Mashhadi
3  * STUDENT ID   :   1084104
4  * ASSIGNMENT#1 :   Functions and Arrays
5  * CLASS        :   CS1B
6  * SECTION     :   MW - 6:30pm - 9:50pm
7  * DUE DATE    :   1/29/2018
8  * *****/
9
10 #include <iostream>
11 #include <iomanip>
12 #include <string>
13 #include <fstream>
14 using namespace std;
15
16 /*****
17  * FUNCTION InputFunction
18  * -----
19  * This function will receive an input file and read 3 sets of data from
20  * the file. It will create three parallel arrays name, Id and the balance.
21  * ==> returns nothing - This will read data from input file.
22  * -----
23  * PRE-CONDITIONS
24  * the following need a defined value pass in
25  *     ifstream &fin : the input file
26  *     fnameAr       : the array for names
27  *     fidsAr        : the array for ids
28  *     fbalanceAr    : the array for balances
29  *     AR_SIZE       : the size of the array
30  * POST-CONDITIONS
31  * ==> Returns nothing - This will read data from input file.
32  *****/
33
34 void InputFunction ( string finFileName, // IN - the name of the file
35                     string fnameAr[],   // IN - read in strings from file to the
36                                     // array
37                     int fidsAr[],       // IN - read in ints from file to the
38                                     // array
39                     double fbalanceAr[], // IN - read in ints from file to the
40                                     // array
41                     const int AR_SIZE) // IN - the size of the array
42 {
43     ifstream fin;
44     int index;
45     index=0;
46
47     fin.open(finFileName.c_str());
48
49     while ( fin && index < AR_SIZE)
50     {
51         getline(fin, fnameAr[index]);
52         fin >> fidsAr[index];
53         fin >> fbalanceAr[index];
54         fin.ignore(1000, '\n');
55         index++;
56     } //END WHILE
57

```

## InputFunction.cpp

```
58
59 //closes input file
60 fin.close();
61 }
62
```

# searchBalance.cpp

```

1 /*****
2  * AURHOR      :   Negin Mashhadi
3  * STUDENT ID  :   1084104
4  * ASSIGNMENT#1 :   Functions and Arrays
5  * CLASS       :   CS1B
6  * SECTION     :   MW - 6:30pm - 9:50pm
7  * DUE DATE    :   1/29/2018
8  * *****/
9 #include <iostream>
10 #include <math.h>
11 using namespace std;
12
13 /*****
14  * FUNCTION SearchBalance
15  * -----
16  *      This function will search for the largest or smallest balance in the
17  *      array of balances based on the user input. it will then return the index
18  *      of the desired balance of the user.
19  *      ==> returns the index of the largest or smallest number.
20  * -----
21  * PRE-CONDITIONS
22  *      the following need a defined value pass in
23  *      fOption      : Option the user enters
24  *      fBalanceAr   : The balance array
25  *      AR_SIZE      : Size of the array
26  * POST-CONDITIONS
27  *      ==> returns the index of the largest or smallest number
28  *****/
29
30 int SearchBalance (int fOption,          // IN - Option input of the user
31                   double fBalanceAr[], // IN - The balance array read from file
32                   const int AR_SIZE)    // IN - The array size
33 {
34     int index;          // CALC          - The index of the array
35     int indexVal;       // CALC & OUT   - The index of the balance wanted
36     double largest;     // CALC          - The largest Balance
37     double smallest;    // CALC          - The smallest Balance
38
39     //Initializing
40     largest = 0;
41     smallest = fBalanceAr[0];
42
43     if( fOption == 1)
44     {
45         for(index = 0; index < AR_SIZE; index++)
46         {
47
48             if (largest < fBalanceAr[index])
49             {
50                 largest = fBalanceAr[index];
51                 indexVal = index;
52             } //END IF FOR LARGEST BALANCE
53
54         } //END FOR LOOP
55
56     }
57     else if(fOption == 2)

```

searchBalance.cpp

```
58 {
59
60     for(index = 0; index < AR_SIZE; index++)
61     {
62         if(fBalanceAr[index] < smallest)
63         {
64             indexVal = index;
65         } //END IF
66     } //END FOR LOOP FOR SMALLEST
67
68 } //END IF ELSE IF
69
70 return indexVal;
71 }
72
```

## SumAndAverage.cpp

```
1 /*****
2  * AURHOR      :   Negin Mashhadi
3  * STUDENT ID   :   1084104
4  * ASSIGNMENT#1 :   Functions and Arrays
5  * CLASS        :   CS1B
6  * SECTION      :   MW - 6:30pm - 9:50pm
7  * DUE DATE     :   1/29/2018
8  * *****/
9 #include "assignment1HeaderFile.h"
10
11 double SumAndAverage (int fOption,          // IN - Option input of the user)
12                      double fBalanceAr[], // IN - The balance array read from
13                      // file
14                      const int AR_SIZE)    // IN - The array size
15 {
16     double valueRequested;                // CALC - the value that will be returned
17                                           // (either sum or avg) based on
18                                           // users input
19
20     double sum;                          // CALC - The sum of all balances
21     double avg;                          // CALC - The average of all balances
22     int index;                          // CALC - The index of the array
23     //Initializing variables
24     sum = 0;
25
26     for(index = 0; index < AR_SIZE; index++)
27     {
28         sum += fBalanceAr[index];
29     }
30
31     avg = sum / double(AR_SIZE);
32
33     if(fOption == 3)
34     {
35         valueRequested = sum;
36     }
37     else if (fOption == 4)
38     {
39
40         valueRequested = avg;
41     }
42
43     return valueRequested;
44 }
45
```

## SearchPerson.cpp

```
1 /*****
2 * AURHOR      :   Negin Mashhadi
3 * STUDENT ID   :   1084104
4 * ASSIGNMENT#1 :   Functions and Arrays
5 * CLASS       :   CS1B
6 * SECTION     :   MW - 6:30pm - 9:50pm
7 * DUE DATE    :   1/29/2018
8 * *****/
9 #include "assignment1HeaderFile.h"
10
11 /*****
12 * FUNCTION SearchName
13 * -----
14 * This function will prompt the user to enter the name of the person they
15 * are looking for. The function will search search for the name through the
16 * name array.
17 * ==> returns the index of the name in the nameAr.
18 * -----
19 * PRE-CONDITIONS
20 * the following need a defined value pass in
21 * fOption      : Option the user enters
22 * fNameAr      : The name array
23 * AR_SIZE      : The size of the array
24 * fSearchName  : The name being searched
25 * POST-CONDITIONS
26 * ==> returns the index of the name in the nameAr.
27 *****/
28
29 int SearchName( string fNameAr[],      // IN - The name array
30                const int AR_SIZE)      // IN - The array size
31
32
33 {
34     bool found;      // CALC      - whether the name being searched is found
35     int index;       // CALC      - Loop Control variable used to search through
36                     //           the file
37     int result;      // CALC & OUT - The index of the name being searched
38     string searchName; // CALC      - Name being searched
39
40     //intitalization
41     result = -1;
42     index = 0;
43     found = false;
44
45     getline(cin, searchName);
46
47     while (index < AR_SIZE && !found)
48     {
49         if(fNameAr[index] == searchName)
50         {
51             found = true;
52             cout << "Found.\n\n";
53             result = index;
54             } // END IF STATMENT
55             index++;
56     } // END WHILE LOOP
57 }
```



SearchPerson.cpp

```
58     if(result == -1)
59     {
60         cout << searchName << " was not found.\n";
61     }//END IF STATEMENT
62
63     return result;
64 }
65
```

## PrintOptions.cpp

```

1 /*****
2  * AURHOR      :   Negin Mashhadi
3  * STUDENT ID   :   1084104
4  * ASSIGNMENT#1 :   Functions and Arrays
5  * CLASS        :   CS1B
6  * SECTION      :   MW - 6:30pm - 9:50pm
7  * DUE DATE     :   1/29/2018
8  * *****/
9 #include "assignment1HeaderFile.h"
10
11 /*****
12  * FUNCTION PrintOptions
13  * -----
14  *   PrintOptions
15  *       This function Will output the results of users search in to the output
16  *   file. It will specifically output the results for find largest balance,
17  *   smallest balance, and person.
18  *       ==> returns nothing - This will output the results.
19  * -----
20  * PRE-CONDITIONS
21  *       the following need a defined value pass in
22  *       fout      : The output file
23  *       fIdsAr     : The id array
24  *       fNameAr    : The name array
25  *       fBalanceAr : The balance array
26  *       index      : The index of the searched item
27  *       fOption    : Option the user enters
28  * POST-CONDITIONS
29  *       ==> returns nothing - This will output the results.
30  *****/
31
32 void PrintOptions(ofstream &fout,           // IN/OUT - output file
33                  int fIdsAr[],              // IN/CALC - the ids array
34                  string fNameAr[],          // IN/CALC - The name array
35                  double fBalanceAr[],       // IN/CALC - The balance array
36                  int index,                 // CALC - The index of the array
37                  int fOption)              // IN - Users choice of the
38                                           // menu
39 {
40     const int IDS_COL = 9;
41     const int NAME_COL = 25;
42     const int BA_COL = 10;
43
44     if(index > -1)
45     {
46         switch(fOption)
47         {
48             case LARGEST: fout << "Largest Balance: " << endl;
49                         break;
50
51             case SMALLEST:
52                 fout << "\nSmallest Balance: " << endl;
53                 break;
54
55             case FINDPERSON:
56                 fout << "\nSearch Name: " << endl;
57                 break;
58         }
59     }
60 }

```

PrintOptions.cpp

```
58     }
59
60     fout << left << setw(IDS_COL) << "ID #";
61     fout << setw(NAME_COL) << "NAME";
62     fout << right << setw(BA_COL) << "BALANCE DUE" << endl;
63     fout << left << setw(IDS_COL) << "----";
64     fout << setw(NAME_COL) << "-----";
65     fout << right << setw(BA_COL) << "-----" << endl;
66     fout << left << setw(IDS_COL) << fIdsAr[index];
67     fout << setw(NAME_COL) << fNameAr[index];
68     fout << '$' << setw(BA_COL) << right << fBalanceAr[index] << endl;
69
70 }
71
72 }
73
```

## PrintSumAndAvg.cpp

```
1 /*****
2 * AURHOR      :   Negin Mashhadi
3 * STUDENT ID   :   1084104
4 * ASSIGNMENT#1 :   Functions and Arrays
5 * CLASS        :   CS1B
6 * SECTION      :   MW - 6:30pm - 9:50pm
7 * DUE DATE     :   1/29/2018
8 * *****/
9 #include "assignment1HeaderFile.h"
10
11 /*****
12 * FUNCTION PrintOptions
13 * -----
14 *   PrintOptions
15 *       This function Will output the results of users search in to the output
16 *       file. It will specifically output the results for sum and average.
17 *       ==> returns nothing - This will output the results.
18 * -----
19 * PRE-CONDITIONS
20 *       the following need a defined value pass in
21 *       fout      : The output file
22 *       fBalance   : The sum or average of the balancec
23 *       prompt     : The output
24 * POST-CONDITIONS
25 *       ==> returns nothing - This will output the results.
26 *****/
27
28 void PrintSumAndAvg(ofstream &fout,      // IN/OUT - output file
29                    double fBalance,      // CALC   - The sum or average
30                    const string prompt)  // OUT    - The prompt for output
31 {
32     fout << prompt;
33     fout << setprecision(2) << fixed;
34     fout << '$' << right << setw(9) << fBalance << endl << endl;
35 }
36
```

## headerFunction.cpp

```

1 /*****
2  * AURHOR      :   Negin Mashhadi
3  * STUDENT ID   :   1084104
4  * ASSIGNMENT#1 :   Functions and Arrays
5  * CLASS        :   CS1B
6  * SECTION      :   MW - 6:30pm - 9:50pm
7  * DUE DATE     :   1/29/2018
8  * *****/
9 #include "assignment1HeaderFile.h"
10 /*****
11  * FUNCTION printHeader
12  * -----
13  * This function receives an assignment name, type and number then outputs the
14  * appropriate class heading.
15  * ==> returns nothing - this function output the class heading.
16  * -----
17  * PRE-CONDITIONS
18  *     the following need a defined value pass in
19  *     output : The output file
20  *     asName: Assignment Name
21  *     asType: Assignment Type
22  *     asNum : Assignment Number
23  *
24  * POST-CONDITIONS
25  *     ==> Returns nothing - this function output the class heading.
26  *****/
27
28 void PrintHeader (ostream &output, // IN/OUT - output file
29                  string  asName, // IN   - assignment Name - used for output
30                  char    asType, // IN   - assignment Type
31                  //        - (LAB or ASSIGN) - used for output
32                  int      asNum) // IN   - assignment Name - used for output
33 {
34 {
35     output << left;
36     output << "*****\n";
37     output << "* PROGRAMMED BY : Negin Mashhadi\n";
38     output << "* " << setw(14) << "STUDENT ID" << ": 1084104\n";
39     output << "* " << setw(14) << "CLASS" << ": CS1B - MW - 6:30pm\n";
40     output << "* ";
41
42     //PROCESSING - This will adjust setws and format appropriately based on if
43     //             this is a lab 'L' or assignment
44
45     if (toupper(asType) == 'L')
46     {
47         output << "LAB # " << setw(9);
48     }
49     else
50     {
51         output << "ASSIGNMENT #" << setw(2);
52     }
53
54     output << asNum << ": " << asName << endl;
55     output << "*****\n\n";
56     output << right<< endl;
57

```

headerFunction.cpp

```
58 }  
59
```

inFile.txt

1 Jean Rousseau  
2 1001 15.50  
3 Steve Woolston  
4 1002 1423.20  
5 Michele Rousseau  
6 1005 52.75  
7 Pete McBride  
8 1007 500.32  
9 Florence Rousseau  
10 1010 1323.33  
11 Lisa Covi  
12 1009 332.35  
13 Don McBride  
14 1003 12.32  
15 Chris Carroll  
16 1008 32.35  
17 Yolanda Agredano  
18 1004 356.00  
19 Sally Sleeper  
20 1006 32.36