Consolidated Assignment 4 Report

This report contains the graded results for the newest of each exercise submitted to
the assignment checker prior to 4/29/2020 5:46:17 PM PDT.

Student Name: Shaun Chemplavil
Student ID: U08713628
Contact e-mail: shaun.chemplavil@gmail.com
C/C++ Programming I (Section 146359)

Submitted:
    Exercise 0: 4/18/2020 7:32:26 AM PDT
    Exercise 1: 4/19/2020 8:09:21 AM PDT
    Exercise 2: 4/23/2020 1:13:29 PM PDT
    Exercise 3: 4/19/2020 8:11:58 AM PDT
    Exercise 4: 4/16/2020 5:17:19 PM PDT

Score (out of 20 possible): ___19___   Please see my comments in your code.

THIS WAS SENT FROM A NOTIFICATION-ONLY ADDRESS THAT CANNOT ACCEPT INCOMING MAIL.
For help please contact the instructor at the email address provided on the
"Announcements" page of the course website. The assignment checker DOES NOT GRADE your
submissions but merely reports on issues so you can correct them and resubmit, thereby
avoiding unnecessary credit loss. ALL GRADING IS DONE MANUALLY BY THE INSTRUCTOR after
the assignment deadline based solely upon the NEWEST submission of each exercise. BE
WARY of correcting minor issues after the deadline because a late deduction will
usually be much greater than a minor issue deduction.

    From: Shaun Chemplavil <mailto:shaun.chemplavil@gmail.com>
    Subject: C1A4E0_U08713628
    Submitted: 4/18/2020 7:32:26 AM PDT
    Course: C/C++ Programming I (Section 146359)
    Student's name: Shaun Chemplavil
    Contact email: shaun.chemplavil@gmail.com
    Student ID: U08713628
    Assignment 4, Exercise 0
    Exercise point value: 6
    File submitted:
       C1A4E0_Quiz.txt

NOTE: The assignment checker does not check the correctness of quiz answers for this
assignment.

Your submission has been accepted and will be graded manually by the instructor. You
may resubmit it as many times as you wish before the assignment deadline. BE WARY of
correcting minor issues after the deadline because a late deduction will usually be
much greater than a minor issue deduction.

Shaun Chemplavil U08713628
shaun.chemplavil@gmail.com
C/C++ Programming I : Fundamental Programming Concepts
146359 Raymond L. Mitchell, Jr., M.S.
04/18/2020
C1A4E0_Quiz.txt
Answers to Quiz

1. D
2. C
3. D **<---E**
4. A
5. E
6. A

THIS WAS SENT FROM A NOTIFICATION-ONLY ADDRESS THAT CANNOT ACCEPT INCOMING MAIL.
For help please contact the instructor at the email address provided on the
"Announcements" page of the course website. The assignment checker DOES NOT GRADE your
submissions but merely reports on issues so you can correct them and resubmit, thereby
avoiding unnecessary credit loss. ALL GRADING IS DONE MANUALLY BY THE INSTRUCTOR after
the assignment deadline based solely upon the NEWEST submission of each exercise. BE
WARY of correcting minor issues after the deadline because a late deduction will
usually be much greater than a minor issue deduction.

    From: Shaun Chemplavil <mailto:shaun.chemplavil@gmail.com>
    Subject: C1A4E1_U08713628
    Submitted: 4/19/2020 8:09:21 AM PDT
    Course: C/C++ Programming I (Section 146359)
    Student's name: Shaun Chemplavil
    Contact email: shaun.chemplavil@gmail.com
    Student ID: U08713628
    Assignment 4, Exercise 1
    Exercise point value: 3
    Files submitted:
        C1A4E1_main.c
        C1A4E1_ComputeMaximum.c
        C1A4E1_ComputeMinimum.c

"Compile-time" results:
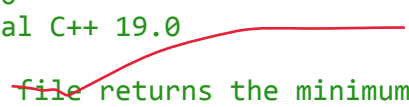   No "compile-time" issues;
"Run-time" results:
    Program ran - No errors detected during preliminary testing (SEE ATTACHMENT);

```
 1   //
 2   // Shaun Chemplavil U08713628
 3   // shaun.chemplavil@gmail.com
 4   // C / C++ Programming I : Fundamental Programming Concepts
 5   // 146359 Raymond L. Mitchell Jr.
 6   // 04 / 19 / 2020
 7   // C1A4E1_ComputeMaximum.c
 8   // Win10
 9   // Visual C++ 19.0
10   //
11   // This file returns the maximum of two type double arguments
12   //
13
14   double ComputeMaximum(double input1, double input2)
15   {
16       return((input1 > input2) ? input1 : input2);
17   }
```

A file cannot return anything.

```
 1  //
 2  // Shaun Chemplavil U08713628
 3  // shaun.chemplavil@gmail.com
 4  // C / C++ Programming I : Fundamental Programming Concepts
 5  // 146359 Raymond L. Mitchell Jr.
 6  // 04 / 19 / 2020
 7  // C1A4E1_ComputeMinimum.c
 8  // Win10
 9  // Visual C++ 19.0
10  //
11  // This file returns the minimum of two type double arguments
12  //
13
14  double ComputeMinimum(double input1, double input2)
15  {
16      return((input1 < input2) ? input1 : input2);
17  }
```

A file cannot return anything.

```c
 1  //
 2  // Shaun Chemplavil U08713628
 3  // shaun.chemplavil@gmail.com
 4  // C / C++ Programming I : Fundamental Programming Concepts
 5  // 146359 Raymond L. Mitchell Jr.
 6  // 04 / 19 / 2020
 7  // C1A4E1_main.c
 8  // Win10
 9  // Visual C++ 19.0
10  //
11  // This file will take two decimal numbers and display their
12  //  maximum and minimum values
13  //
14
15  #include <stdio.h>
16
17  double ComputeMinimum(double input1, double input2);
18  double ComputeMaximum(double input1, double input2);
19
20  int main(void)
21  {
22      double input1, input2;
23
24      // Request and Store User Input
25      printf("Enter two space-separated decimal values on the same line: ");
26      scanf("%lg%lg", &input1, &input2);
27
28      // Display Minimum and Maximum values
29      printf(
30          "ComputeMinimum(%lg,%lg) returned %lg\n"
31          "ComputeMaximum(%lg,%lg) returned %lg\n",
32          input1, input2, ComputeMinimum(input1, input2),
33          input1, input2, ComputeMaximum(input1, input2));
34
35      return 0;
36  }
```

A file cannot return anything.

```
********** C1 ASSIGNMENT 4 EXERCISE 1 AUTOMATIC PROGRAM RUN RESULTS **********

************* THE RESULTS BELOW HAVE BEEN PARTIALLY CHECKED AND **************
*************   NO ERRORS WERE FOUND.  HOWEVER, THIS DOES NOT   **************
************* NECESSARILY MEAN THAT THERE ARE NO ERRORS.  THE   **************
*************   THE INSTRUCTOR WILL DO A MORE THOROUGH CHECK     **************
*************            DURING MANUAL GRADING.                 **************


---------------------------- START OF 1ST RUN ----------------------------


Enter two space-separated decimal values on the same line: 3 3
ComputeMinimum(3,3) returned 3
ComputeMaximum(3,3) returned 3



---------------------------- END OF 1ST RUN ----------------------------

---------------------------- START OF 2ND RUN ----------------------------


Enter two space-separated decimal values on the same line: -7.98 7.13
ComputeMinimum(-7.98,7.13) returned -7.98
ComputeMaximum(-7.98,7.13) returned 7.13



---------------------------- END OF 2ND RUN ----------------------------

---------------------------- START OF 3RD RUN ----------------------------


Enter two space-separated decimal values on the same line: 2000.45 0
ComputeMinimum(2000.45,0) returned 0
ComputeMaximum(2000.45,0) returned 2000.45



---------------------------- END OF 3RD RUN ----------------------------

---------------------------- START OF 4TH RUN ----------------------------


Enter two space-separated decimal values on the same line: 54e-2 86e-1
ComputeMinimum(0.54,8.6) returned 0.54
ComputeMaximum(0.54,8.6) returned 8.6



---------------------------- END OF 4TH RUN ----------------------------

---------------------------- START OF 5TH RUN ----------------------------


Enter two space-separated decimal values on the same line: 86e-1 54e-2
ComputeMinimum(8.6,0.54) returned 0.54
ComputeMaximum(8.6,0.54) returned 8.6



---------------------------- END OF 5TH RUN ----------------------------

---------------------------- START OF 6TH RUN ----------------------------


Enter two space-separated decimal values on the same line: -0 0
ComputeMinimum(-0,0) returned 0
ComputeMaximum(-0,0) returned 0
```

------------------------------ END OF 6TH RUN ------------------------------

THIS WAS SENT FROM A NOTIFICATION-ONLY ADDRESS THAT CANNOT ACCEPT INCOMING MAIL.
For help please contact the instructor at the email address provided on the
"Announcements" page of the course website. The assignment checker DOES NOT GRADE your
submissions but merely reports on issues so you can correct them and resubmit, thereby
avoiding unnecessary credit loss. ALL GRADING IS DONE MANUALLY BY THE INSTRUCTOR after
the assignment deadline based solely upon the NEWEST submission of each exercise. BE
WARY of correcting minor issues after the deadline because a late deduction will
usually be much greater than a minor issue deduction.

    From: <mailto:shaun.chemplavil@gmail.com>
    Subject: C1A4E2_U08713628
    Submitted: 4/23/2020 1:13:29 PM PDT
    Course: C/C++ Programming I (Section 146359)
    Student's name: Shaun Chemplavil
    Contact email: shaun.chemplavil@gmail.com
    Student ID: U08713628
    Assignment 4, Exercise 2
    Exercise point value: 4
    Files submitted:
       C1A4E2_PrintLines-2.cpp
       C1A4E2_PrintLines-3.cpp
       C1A4E2_main.cpp
       C1A4E2_PrintLines-0.cpp
       C1A4E2_PrintLines-1.cpp
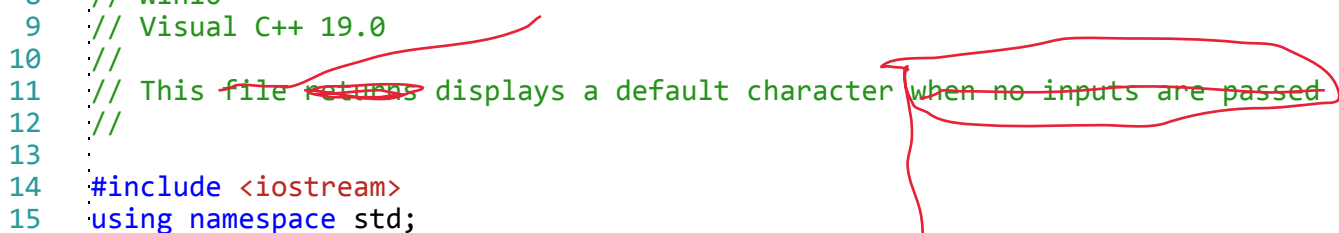
"Compile-time" results:
    No "compile-time" issues;
"Run-time" results:
    Program ran - No errors detected during preliminary testing (SEE ATTACHMENT);

```
 1   //
 2   // Shaun Chemplavil U08713628
 3   // shaun.chemplavil@gmail.com
 4   // C / C++ Programming I : Fundamental Programming Concepts
 5   // 146359 Raymond L. Mitchell Jr.
 6   // 04 / 23 / 2020
 7   // C1A4E2_main.cpp
 8   // Win10
 9   // Visual C++ 19.0                    A file cannot take inputs or call functions.
10   //
11   // This file will take user inputs varies calls to an overloaded function
12   //
13
14   #include <iostream>
15   using namespace std;
16
17   void PrintLines(int charsToDisplay, int numberOfCharacters, int numberOfLines);
18   void PrintLines(int charsToDisplay, int numberOfCharacters);
19   void PrintLines(int charsToDisplay);
20   void PrintLines(void);
21
22   // Define the number of times we repeat the main body
23   const int REPEAT = 2;
24
25   int main()
26   {
27       // Repeat main Block REPEAT times
28       for (int reps = 0; reps < REPEAT; reps++)
29       {
30           char inputChar;
31           int numChars, numLines;
32
33           // Request and Store User Input
34           cout <<
35               "Enter the following space separated items on the same line:\n"
36               "Character to display.\n "
37               "Number of times to display this character.\n"
38               "Number of lines to display: ";
39
40           cin >> inputChar >> numChars >> numLines;
41
42           // Because the first input expects to be type int, we must type cast it
43           PrintLines(int(inputChar), numChars, numLines);
44           PrintLines(int(inputChar), numChars);
45           PrintLines(int(inputChar));
46           // Output 'Z' when no inputs are passed
47           PrintLines();
48       }
49       return 0;
50   }
```

```
 1  //
 2  // Shaun Chemplavil U08713628
 3  // shaun.chemplavil@gmail.com
 4  // C / C++ Programming I : Fundamental Programming Concepts
 5  // 146359 Raymond L. Mitchell Jr.
 6  // 04 / 23 / 2020
 7  // C1A4E2_PrintLines-0.cpp
 8  // Win10
 9  // Visual C++ 19.0
10  //
11  // This file returns displays a default character when no inputs are passed
12  //
13
14  #include <iostream>
15  using namespace std;
16
17  const char DEFAULT_CHAR = 'Z';
18
19  void PrintLines()
20  {
21      cout << DEFAULT_CHAR << "\n";
22  }
```

It is impossible to pass any arguments to it.

```
 1  //
 2  // Shaun Chemplavil U08713628
 3  // shaun.chemplavil@gmail.com
 4  // C / C++ Programming I : Fundamental Programming Concepts
 5  // 146359 Raymond L. Mitchell Jr.
 6  // 04 / 23 / 2020
 7  // C1A4E2_PrintLines-1.cpp
 8  // Win10
 9  // Visual C++ 19.0
10  //
11  // This file returns takes user character input and displays it once
12  //
13
14  #include <iostream>
15  using namespace std;
16
17  void PrintLines(int characterToDisplay)
18  {
19      cout << char(characterToDisplay) << "\n";
20  }
```

Do you know the meaning of the word return in the context of a function?

```cpp
 1  //
 2  // Shaun Chemplavil U08713628
 3  // shaun.chemplavil@gmail.com
 4  // C / C++ Programming I : Fundamental Programming Concepts
 5  // 146359 Raymond L. Mitchell Jr.
 6  // 04 / 23 / 2020
 7  // C1A4E2_PrintLines-2.cpp
 8  // Win10
 9  // Visual C++ 19.0
10  //
11  // This file returns takes user character input and displays the number of times
12  //  designated by the second input
13  //
14
15  #include <iostream>
16  using namespace std;
17
18  void PrintLines(int charsToDisplay, int numberOfCharacters)
19  {
20      // Loop tracking the number of characters to display
21      for (int charCnt = 0; charCnt < numberOfCharacters; charCnt++)
22      {
23          cout << char(charsToDisplay);
24      }
25      cout << "\n";
26  }
```

```cpp
//
// Shaun Chemplavil U08713628
// shaun.chemplavil@gmail.com
// C / C++ Programming I : Fundamental Programming Concepts
// 146359 Raymond L. Mitchell Jr.
// 04 / 23 / 2020
// C1A4E2_PrintLines-3.cpp
// Win10
// Visual C++ 19.0
//
// This file returns takes user character input and displays the number of times
//  designated by the second input, and repeats this according to the third
//  input
//

#include <iostream>
using namespace std;

void PrintLines(int charsToDisplay, int numberOfCharacters, int numberOfLines)
{
    // Loop tracking the number of lines to display
    for (int lineCnt = 0; lineCnt < numberOfLines; lineCnt++)
    {
        // Loop tracking the number of characters to display
        for (int charCnt = 0; charCnt < numberOfCharacters; charCnt++)
        {
            cout << char(charsToDisplay);
        }
        cout << "\n";
    }
}
```

********** C1 ASSIGNMENT 4 EXERCISE 2 AUTOMATIC PROGRAM RUN RESULTS **********

************* THE RESULTS BELOW HAVE BEEN PARTIALLY CHECKED AND **************
*************   NO ERRORS WERE FOUND. HOWEVER, THIS DOES NOT   **************
************* NECESSARILY MEAN THAT THERE ARE NO ERRORS. THE **************
************* INSTRUCTOR WILL DO A MORE THOROUGH CHECK DURING **************
*************   MANUAL GRADING.   **************

---------------------------- START OF 1ST RUN ----------------------------

Enter the following space separated items on the same line:
Character to display.
 Number of times to display this character.
Number of lines to display: J 2 3
JJ
JJ
JJ
JJ
J
Z
Enter the following space separated items on the same line:
Character to display.
 Number of times to display this character.
Number of lines to display: @ 25 5
@@@@@@@@@@@@@@@@@@@@@@@@@
@@@@@@@@@@@@@@@@@@@@@@@@@
@@@@@@@@@@@@@@@@@@@@@@@@@
@@@@@@@@@@@@@@@@@@@@@@@@@
@@@@@@@@@@@@@@@@@@@@@@@@@
@@@@@@@@@@@@@@@@@@@@@@@@@
@
Z

---------------------------- END OF 1ST RUN ----------------------------

---------------------------- START OF 2ND RUN ----------------------------

Enter the following space separated items on the same line:
Character to display.
 Number of times to display this character.
Number of lines to display: X 6 15
XXXXXX
XXXXXX
XXXXXX
XXXXXX
XXXXXX
XXXXXX
XXXXXX
XXXXXX
XXXXXX
XXXXXX
XXXXXX
XXXXXX
XXXXXX
XXXXXX
XXXXXX
X
Z

Enter the following space separated items on the same line:
Character to display.
 Number of times to display this character.
Number of lines to display: % 4 0
%%%
%
Z

----------------------------- END OF 2ND RUN -------------------------------

---------------------------- START OF 3RD RUN ------------------------------

Enter the following space separated items on the same line:
Character to display.
 Number of times to display this character.
Number of lines to display: b 0 4

b
Z
Enter the following space separated items on the same line:
Character to display.
 Number of times to display this character.
Number of lines to display: $ 0 0

$
Z

----------------------------- END OF 3RD RUN -------------------------------

THIS WAS SENT FROM A NOTIFICATION-ONLY ADDRESS THAT CANNOT ACCEPT INCOMING MAIL.
For help please contact the instructor at the email address provided on the
"Announcements" page of the course website. The assignment checker DOES NOT GRADE your
submissions but merely reports on issues so you can correct them and resubmit, thereby
avoiding unnecessary credit loss. ALL GRADING IS DONE MANUALLY BY THE INSTRUCTOR after
the assignment deadline based solely upon the NEWEST submission of each exercise. BE
WARY of correcting minor issues after the deadline because a late deduction will
usually be much greater than a minor issue deduction.

    From: Shaun Chemplavil <mailto:shaun.chemplavil@gmail.com>
    Subject: C1A4E3_U08713628
    Submitted: 4/19/2020 8:11:58 AM PDT
    Course: C/C++ Programming I (Section 146359)
    Student's name: Shaun Chemplavil
    Contact email: shaun.chemplavil@gmail.com
    Student ID: U08713628
    Assignment 4, Exercise 3
    Exercise point value: 3
    Files submitted:
       C1A4E3_PrintLines.cpp
       C1A4E3_main.cpp

"Compile-time" results:
   No "compile-time" issues;
"Run-time" results:
   Program ran - No errors detected during preliminary testing (SEE ATTACHMENT);

```cpp
1   //
2   // Shaun Chemplavil U08713628
3   // shaun.chemplavil@gmail.com
4   // C / C++ Programming I : Fundamental Programming Concepts
5   // 146359 Raymond L. Mitchell Jr.
6   // 04 / 19 / 2020
7   // C1A4E3_main.cpp
8   // Win10
9   // Visual C++ 19.0
10  //
11  // This file will take user inputs varies calls to a function with
12  //  default parameters in prototype function
13
14  #include <iostream>
15  using namespace std;
16
17  void PrintLines(int charToDisp = int('Z'), int numChars = 1, int numLines = 1);
18
19  // Define the number of times we repeat the main body
20  const int REPEAT = 2;
21
22  int main()
23  {
24      // Repeat main Block REPEAT times
25      for (int reps = 0; reps < REPEAT; reps++)
26      {
27          char inputChar;
28          int numChars, numLines;
29
30          // Request and Store User Input
31          cout <<
32              "Enter the following space separated items on the same line:\n"
33              "Character to display.\n "
34              "Number of times to display this character.\n"
35              "Number of lines to display: ";
36
37          cin >> inputChar >> numChars >> numLines;
38
39          // Because the first input expects to be type int, we must type cast it
40          PrintLines(int(inputChar), numChars, numLines);
41          PrintLines(int(inputChar), numChars);
42          PrintLines(int(inputChar));
43          // Output 'Z' when no inputs are passed
44          PrintLines();
45      }
46      return 0;
47  }
```

```
 1  //
 2  // Shaun Chemplavil U08713628
 3  // shaun.chemplavil@gmail.com
 4  // C / C++ Programming I : Fundamental Programming Concepts
 5  // 146359 Raymond L. Mitchell Jr.
 6  // 04 / 19 / 2020
 7  // C1A4E3_PrintLines.cpp
 8  // Win10
 9  // Visual C++ 19.0
10  //
11  // This file returns takes user character input and displays the number of times
12  //  designated by the second input, and repeats this according to the third
13  //  input
14  //
15
16  #include <iostream>
17  using namespace std;
18
19  void PrintLines(int charsToDisplay, int numberOfCharacters, int numberOfLines)
20  {
21      // Loop tracking the number of lines to display
22      for (int lineCnt = 0; lineCnt < numberOfLines; lineCnt++)
23      {
24          // Loop tracking the number of characters to display
25          for (int charCnt = 0; charCnt < numberOfCharacters; charCnt++)
26          {
27              cout << char(charsToDisplay);
28          }
29          cout << "\n";
30      }
31  }
```

********** C1 ASSIGNMENT 4 EXERCISE 3 AUTOMATIC PROGRAM RUN RESULTS **********

************* THE RESULTS BELOW HAVE BEEN PARTIALLY CHECKED AND **************
*************   NO ERRORS WERE FOUND. HOWEVER, THIS DOES NOT   **************
************* NECESSARILY MEAN THAT THERE ARE NO ERRORS. THE   **************
************* THE INSTRUCTOR WILL DO A MORE THOROUGH CHECK   **************
*************   DURING MANUAL GRADING.   **************

---------------------------- START OF 1ST RUN ----------------------------

Enter the following space separated items on the same line:
Character to display.
 Number of times to display this character.
Number of lines to display: J 2 3
JJ
JJ
JJ
JJ
J
Z
Enter the following space separated items on the same line:
Character to display.
 Number of times to display this character.
Number of lines to display: @ 25 5
@@@@@@@@@@@@@@@@@@@@@@@@@
@@@@@@@@@@@@@@@@@@@@@@@@@
@@@@@@@@@@@@@@@@@@@@@@@@@
@@@@@@@@@@@@@@@@@@@@@@@@@
@@@@@@@@@@@@@@@@@@@@@@@@@
@@@@@@@@@@@@@@@@@@@@@@@@@
@
Z

---------------------------- END OF 1ST RUN -----------------------------

---------------------------- START OF 2ND RUN ----------------------------

Enter the following space separated items on the same line:
Character to display.
 Number of times to display this character.
Number of lines to display: X 6 15
XXXXXX
XXXXXX
XXXXXX
XXXXXX
XXXXXX
XXXXXX
XXXXXX
XXXXXX
XXXXXX
XXXXXX
XXXXXX
XXXXXX
XXXXXX
XXXXXX
XXXXXX
X
Z

Enter the following space separated items on the same line:
Character to display.
 Number of times to display this character.
Number of lines to display: % 4 0
%%%
%
Z

---------------------------- END OF 2ND RUN ------------------------------

---------------------------- START OF 3RD RUN ----------------------------

Enter the following space separated items on the same line:
Character to display.
 Number of times to display this character.
Number of lines to display: b 0 4

b
Z
Enter the following space separated items on the same line:
Character to display.
 Number of times to display this character.
Number of lines to display: $ 0 0

$
Z

---------------------------- END OF 3RD RUN ------------------------------

THIS WAS SENT FROM A NOTIFICATION-ONLY ADDRESS THAT CANNOT ACCEPT INCOMING MAIL.
For help please contact the instructor at the email address provided on the
"Announcements" page of the course website. The assignment checker DOES NOT GRADE your
submissions but merely reports on issues so you can correct them and resubmit, thereby
avoiding unnecessary credit loss. ALL GRADING IS DONE MANUALLY BY THE INSTRUCTOR after
the assignment deadline based solely upon the NEWEST submission of each exercise. BE
WARY of correcting minor issues after the deadline because a late deduction will
usually be much greater than a minor issue deduction.

    From: Shaun Chemplavil <mailto:shaun.chemplavil@gmail.com>
    Subject: C1A4E4_U08713628
    Submitted: 4/16/2020 5:17:19 PM PDT
    Course: C/C++ Programming I (Section 146359)
    Student's name: Shaun Chemplavil
    Contact email: shaun.chemplavil@gmail.com
    Student ID: U08713628
    Assignment 4, Exercise 4
    Exercise point value: 4
    Files submitted:
        C1A4E4_main.cpp
        C1A4E4_MaxOf.h

"Compile-time" results:
    No "compile-time" issues;
"Run-time" results:
    Program ran - No errors detected during preliminary testing (SEE ATTACHMENT);

```
 1   //
 2   // Shaun Chemplavil U08713628
 3   // shaun.chemplavil@gmail.com
 4   // C / C++ Programming I : Fundamental Programming Concepts
 5   // 146359 Raymond L. Mitchell Jr.
 6   // 04 / 16 / 2020
 7   // C1A4E4_MaxOf.h
 8   // Win10
 9   // Visual C++ 19.0
10   //
11   // Header file containing prototype of C1A4E4 Functions
12   //
13
14   // include guard
15   #ifndef  C1A4E4_MAXOF_H
16   #define C1A4E4_MAXOF_H
17
18   // Macro that Returns Maximum of 2 Inputs
19   #define mMaxOf2(in1, in2) ((in1) > (in2) ? (in1) : (in2))
20   // Macro that Returns Maximum of 3 Inputs
21   #define mMaxOf3(in1, in2, in3) (mMaxOf2(mMaxOf2((in1), (in2)), (in3)))
22
23   // Inline function that Returns Maximum of 2 Inputs
24   inline long double fMaxOf2(long double in1, long double in2)
25   {
26       return in1 > in2 ? in1 : in2;
27   }
28
29   // Inline function that Returns Maximum of 3 Inputs
30   inline long double fMaxOf3(long double in1, long double in2, long double in3)
31   {
32       return fMaxOf2(fMaxOf2(in1, in2), in3);
33   }
34
35   #endif
```

```cpp
1   //
2   // Shaun Chemplavil U08713628
3   // shaun.chemplavil@gmail.com
4   // C / C++ Programming I : Fundamental Programming Concepts
5   // 146359 Raymond L. Mitchell Jr.
6   // 04 / 16 / 2020
7   // C1A4E4_main.cpp
8   // Win10
9   // Visual C++ 19.0
10  //
11  // This file will find the maximum of 3 user input values and return the maximum
12  // value using an inline function and a macro
13  //
14
15  #include <iostream>
16  using namespace std;
17
18  // Include header containing function prototypes
19  #include "C1A4E4_MaxOf.h"
20
21  int main()
22  {
23      long double in1, in2, in3, macroOut, inlineOut;
24
25      // Request User Input
26      cout << "Enter the three space-separated decimal values on the same line:\n";
27
28      cin >> in1 >> in2 >> in3;
29
30      // Find max value using a macro
31      macroOut = mMaxOf3(in1, in2, in3);
32
33      // Find max value using an inline function
34      inlineOut = fMaxOf3(in1, in2, in3);
35
36      // Display Results...
37      cout <<
38          "mMaxOf3(" << in1 << "," << in2 << "," << in3 << ")"
39          " returned " << macroOut << "\n"
40          "fMaxOf3(" << in1 << "," << in2 << "," << in3 << ")"
41          " returned " << inlineOut << "\n";
42
43      return 0;
44  }
```

This function returns 0.

********** C1 ASSIGNMENT 4 EXERCISE 4 AUTOMATIC PROGRAM RUN RESULTS **********

************* THE RESULTS BELOW HAVE BEEN PARTIALLY CHECKED AND *************
*************   NO ERRORS WERE FOUND. HOWEVER, THIS DOES NOT   *************
************* NECESSARILY MEAN THAT THERE ARE NO ERRORS. THE *************
*************   THE INSTRUCTOR WILL DO A MORE THOROUGH CHECK   *************
*************   DURING MANUAL GRADING.   *************

---------------------------- START OF 1ST RUN ----------------------------

Enter the three space-separated decimal values on the same line:
3 3 3
mMaxOf3(3,3,3) returned 3
fMaxOf3(3,3,3) returned 3

---------------------------- END OF 1ST RUN ----------------------------

---------------------------- START OF 2ND RUN ----------------------------

Enter the three space-separated decimal values on the same line:
-7.98 7.13 -18.2
mMaxOf3(-7.98,7.13,-18.2) returned 7.13
fMaxOf3(-7.98,7.13,-18.2) returned 7.13

---------------------------- END OF 2ND RUN ----------------------------

---------------------------- START OF 3RD RUN ----------------------------

Enter the three space-separated decimal values on the same line:
2000.45 0 2000.46
mMaxOf3(2000.45,0,2000.46) returned 2000.46
fMaxOf3(2000.45,0,2000.46) returned 2000.46

---------------------------- END OF 3RD RUN ----------------------------

---------------------------- START OF 4TH RUN ----------------------------

Enter the three space-separated decimal values on the same line:
33.3 54e-2 86e-1
mMaxOf3(33.3,0.54,8.6) returned 33.3
fMaxOf3(33.3,0.54,8.6) returned 33.3

---------------------------- END OF 4TH RUN ----------------------------

---------------------------- START OF 5TH RUN ----------------------------

Enter the three space-separated decimal values on the same line:
86e-1 86e-1 54e-2
mMaxOf3(8.6,8.6,0.54) returned 8.6
fMaxOf3(8.6,8.6,0.54) returned 8.6

---------------------------- END OF 5TH RUN ----------------------------

---------------------------- START OF 6TH RUN ----------------------------

Enter the three space-separated decimal values on the same line:
-0 0 0
mMaxOf3(-0,0,0) returned 0
fMaxOf3(-0,0,0) returned 0

------------------------------ END OF 6TH RUN ------------------------------