60-140-1 and 60-140-2 ASSIGNMENT #4 SOLUTION
Handed Out:Thurs. Oct 22, 2015 for (60-140-01 and 60-140-02)
Due: Thurs Oct 29, 2015 for (60-140-01 and 60-140-02)
Total: 50 marks

Objective of Assignment : To continue to write a C program to solve a problem using functions and function calls (including call-by-reference parameters and global variables), but no decision or repetition instructions.

Scope : Assignment covers materials up until end of chapter 4.

Other Things to learn from Assignment : How to make function calls using **call-by-value** parameters for input parameters and **call-by-reference** parameters for all output parameters. Also, how make function calls with no **parameters where all variables are global**.

DO NOT USE GLOBAL VARIABLES IN THE FIRST PROBLEM SOLUTION.

Important: Do not forget to type in your ***full name***, ***student number***, ***lecture section number,*** ***lab section number*** and ***date*** in BOTH the algorithm and source C program files.

Electronic Assignment Submission:

03-60-140-1 students: email script file to cs140_01@cs.uwindsor.ca with subject including: Name, student id, lecture section, lab [section], assignment #4 (in the subject of the mail submission of script file).

03-60-140-2 students: email script file to cs140_02@cs.uwindsor.ca with subject including: Name, student id, lecture section, lab [section], assignment #4 (in the subject of the mail submission of script file).

*Only the assignments currently due that are submitted to this site within two days before and by the due date, are retrieved for marking. Others are deleted soon after.

Problem :

Given the conversion rate between degree Celsius (°C) and degree Fahrenheit (°F) is °C = 5.0/9.0 (°F- 32.0), use two functions **Calc_Celsius** (which returns the Celsius equivalent of a Fahrenheit temperature) and **Calc_Fahr** (which returns the Fahrenheit equivalent of a Celsius temperature) to write two programs (one with call-by-reference parameters and the other with global variables), each of which prints

i) the Celsius temperatures of three Fahrenheit temperatures entered at the keyboard by the user as input data, and

ii) the Fahrenheit equivalents of three Celsius temperatures entered at the keyboard by the user as input data.

You should print lines of output in a neat tabular form as shown in the sample output, by calling the functions and without using repetition instructions.

Note that the above formula implies that °F = (9.0 * °C/5.0) + 32.0.

Rules to follow:

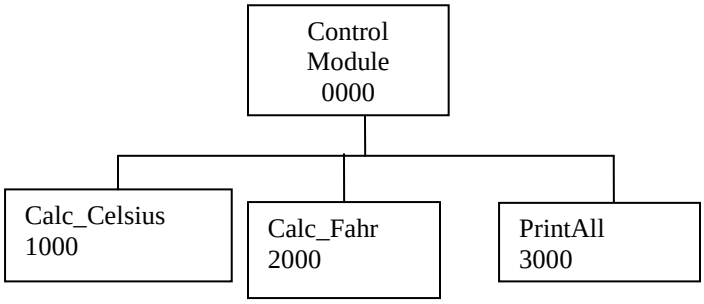
You are required to write two C programs using top-down design approach (that is, using functions as specified in the given structure chart) for the problem described below. The requirements of the two programs are as follows:

1) The first program **should not use any global variables**, must pass a**ll input parameters using call-by-value parameter**s and must pass all **function result parameters using call-by-reference parameters**.
2) The second program should declare all main program input and output variables globally and should not pass any parameters in the function calls. Note that a function can still declare needed local variables.
3) You are also required to show tracing (**only step 6 and not all 6 problem solving steps**) of your two programs with test data to test for correctness of your solutions.

All input data must be read in the control or main module and all results printed by another function dedicated to printing. **Use the following structure chart for solving the problem.**

******

All input data should be read in the control module.
Cacl_Fahr module should calculate the Fahrenheit equivalents.
Calc_Celsius module should be used to calculate the Celsius equivalents.
PrintAll should print all data and tables.



Sample Input:

```
Please type the three fahrenheit temperatures to be converted to
Celsius:
68  -40  100
Please type the three Celsius temperatures to be converted to
Fahrenheit:
0  -40  20
```

Sample Output:

```
 The Temperature Conversions from Fahrenheit to Celsius are

   Fahrenheit      | Celsius
  *********************************
```

```
  68.00           | 20.00
  -40.00          | -40.00
  100.00          | 37.78
 *****
The Temperature Conversions from Celsius to Fahrenheit are
  Celsius         | Fahrenheit
 ********************************
  0.00            | 32.00
  -40.00          | -40.00
  20.00           | 68.00
```

Thank You for Using the UWin_60-140 Temperature Conversion System!!
Have a Nice Day and Come Again !!

****

Type your tracing of each of your two programs (for testing and verification) for solving this problem in each of the text files called userid_asn4_prog1.txt and userid_asn4_prog2.txt. Then, type each of the C program solutions into a source file called your userid_asn4_prog1.c and userid_asn4_prog2.c. Compile your C program with cc userid_asn4_prog1.c or cc userid_asn4_prog1.c. Then, run (execute it with ./a.out). When your program is running with no errors, then, create a script file called userid_scrpt41.txt and userid_scrpt42.txt and hand in your script files for marking. Your script file 1 should show your tracing of source program 1, source program 1, the compilation of your program 1, the running of your program 1 with the input and output data shown and your script file 2 should show your tracing of source program 2, source program 2, the compilation of your program 2, the running of your program 2 with the input and output data shown. The script files can be created as follows:

script userid_script41.txt
cat userid_asn4_prog1.txt
cat userid_asn4_prog1.c
cc userid_asn4_prog1.c
./a.out
(when prompted, type in the input data in the correct order)
exit

**

script userid_script42.txt
cat userid_asn4_prog2.txt
cat userid_asn4_prog2.c
cc userid_asn4_prog2.c
./a.out
(when prompted, type in the input data in the correct order)
exit

Marking Scheme :

1. Correct Program Solution for program 1 with call-by-reference parameters
   (3 for variable declarations, 2 for function prototypes, 2 for function calls,
   6 for function definitions, 3 for reading and printing, 4 for correct logic) ---(20marks)

2. Correct Program Solution for program 2 with global variables
   (3 for variable declarations, 2 for function prototypes, 2 for function calls,
   6 for function definitions,3 for reading and printing, 4 for correct logic) ----(20marks)
3. Hand tracing the programs for testing and evaluation (step 6)
   (5 marks for correct tracing of program 1, and 5 marks for correct
   tracing of program 2) ------(10 marks)

SOLUTION

The tracing of each of the two programs, their programs and execution of the program with input and output data are shown below in the contents of the two script files.

Solution of Program 1

```
Script started on Wed 30 Sep 2015 02:32:23 PM EDT
cezeife@charlie:~/fall15/assignmt$ cat cezeife_asn4_prog1.txt

Step 6: Test and Verification
The execution of the program begins in the main module, where the first
Three input data are read for Fahrenheit as Fahren1 = 68, Fahren2= -40,
Fahren3 = 100.  Then, the second set of input data are read for
Celsius1 = 0, Celsius2 = -40 and Celsius3 = 20. The next instruction
Executed is Calc_Celsius(Fahren1, &Celsius_out1);
This instruction calls the function Calc_Celsius(68, &Celsius_out1) by
passing the
call-by-value input parameter 68 to this function Calc_Celsius and
call-by-reference
output parameter, the address of &Celsius_out1. Passing the address of
Celsius_out1 gives the function Calc_Celsius the permission to directly
write the computed Celsius conversion of Fahren1 = 68 into the main
function's
variable Celsius_out1. Inside the function Celsius, the formal
parameter,
Fahren takes the actual parameter value of 68 for Fahren1. And the
instructions of this function Celsius are executed as follows.
*Celsius = 5.0/9.0 * (Fahren- 32.0) = 5.0/9.0 * (68 - 32.0) = 20.
This value is written directly to main's variable Celsius_out1 as  20.
The function Calc_Celsius is called again a second time with the
instruction
Calc_Celsius(Fahren2, &Celsius_out2); This passes a new actual
parameter of
Fahren2 = -40 into Celsius function and allows the result of the
computation
to be saved directly through call-by-reference parameter into the
main's
variable, Celsius_out2 whose address is passed as second parameter.
In Celsius function, the instruction *Celsius = 5.0/9.0 * (Fahren-
32.0) =
```

5.0/9.0 * (-40 - 32.0) = -40. The value -40 is returned to main at the point
of call so that *Celsius_out2 = -40. The third function call passes Fahren3=100 to Fahren and passes &Celsius_out3 to *Celsius= 5.0/9.0 * (Fahren- 32.0) = 5.0/9.0 * (100 - 32) = 37.8 so that *Celsius_out3=37.8 in main.
The next three sequence of instructions cause the function, Fahrenheit to be called by main with the parameters 0 for Celsius1 in the first time, -40 for Celsius2 in the second time and 20 for Celsius3 in the third time. The instructions are:
Calc_Fahr(Celsius1, &Fahren_out1);
Calc_Fahr(Celsius2, &Fahren_out2);
Calc_Fahr(Celsius3, &Fahren_out3);
Inside the function Fahrenheit, the instruction
*Fahren = (9.0 * Celsius)/5.0 + 32;
is evaluated each time to save directly to the main variables the consecutive
values *Fahren_out1 = (9.0 * Celsius)/5.0 + 32 = 9.0 * 0 /5.0 + 32 = 32.
Next is *Fahren_out2 = (9.0 * Celsius)/5.0 + 32 = 9.0 * -40/5.0 + 32 = -40.
The third stores in Fahren_out3 the value *Fahren = (9.0 * Celsius)/5.0 + 32 = 9.0 * 20/5.0 + 32 = 68.
The function PrintAll is called by passing all these input and output variables
(the twelf variables) as call-by-value parameters and printf instructions
are used to print the results in a table as shown in the sample output in the
script file.

END.

cezeife@charlie:~/fall15/assignmt$ cat cezeife_asn4_prof□□□[Kg1.c

```
#include <stdio.h>
 /*
 Problem: Given the conversion rate between degree Celsius (C) and degree Fa
hrenheit (F) is C = 5.0/9.0 (F- 32.0), use two functions Calc_Celsius (wh
ich returns the Celsius equivalent of a Fahrenheit temperature) and Calc_Fahrenheit (
which returns the Fahrenheit equivalent of a Celsius temperature) to write an al
gorithm and a program, which prints
iii)    the Celsius temperatures of  three Fahrenheit temperatures entered at th
e keyboard by the user for input data, and
iv)     the Fahrenheit equivalents of three Celsius temperatures entered at the
```

```
keyboard by the user for input data.

The problem solution prints lines of output in a neat tabular form as
shown in t
he sample output, by calling the functions and without using repetition
instruct
ions.

This solution uses call-by-reference parameters for all function output
parameters, but uses call-by-value parameters for all function input
parameters.

*/
#include <stdio.h>

/* Three function prototypes                                  */
void Calc_Celsius (float, float *);
void Calc_Fahr(float, float *);
void PrintAll(float, float, float, float, float, float, float, float,
float, float, float, float);

/* This is the main Module 0000 which calls modules Celsius and
Fahrenheit
*/
int main(void)
{
float Fahren1, Fahren2, Fahren3, Celsius1, Celsius2, Celsius3;

 float  Fahren_out1, Fahren_out2, Fahren_out3, Celsius_out1,
Celsius_out2, Celsius_out3;
/* Now the sequence of instructions   */
printf("Please type the three fahrenheit temperatures to be converted
to Celsius:\n");
scanf("%f %f %f", &Fahren1, &Fahren2, &Fahren3);
printf("Please type the three Celsius temperatures to be converted to
Fahrenheit:\n");
scanf("%f %f %f", &Celsius1, &Celsius2, &Celsius3);
Calc_Celsius(Fahren1, &Celsius_out1);
Calc_Celsius(Fahren2, &Celsius_out2);
Calc_Celsius(Fahren3, &Celsius_out3);
Calc_Fahr(Celsius1, &Fahren_out1);
Calc_Fahr(Celsius2, &Fahren_out2);
Calc_Fahr(Celsius3, &Fahren_out3);
PrintAll(Fahren1, Fahren2, Fahren3, Celsius1, Celsius2, Celsius3,
Fahren_out1, Fahren_out2, Fahren_out3, Celsius_out1,  Celsius_out2,
Celsius_out3);

return 0;
}

/***** The function definition for function Celsius which accepts a
Temperature in Fahrenheit and prints its equivalent in Celsius.  This
called
module 1000
```

```
*/
void Calc_Celsius(float Fahren, float *Celsius)
{
*Celsius = 5.0/9.0 * (Fahren- 32.0);
}

/***** The function definition for function Celsius which accepts a
   Temperature in Fahrenheit and prints its equivalent in Celsius.
This called
module 2000

*/

void Calc_Fahr(float Celsius, float *Fahren)
{
  *Fahren = (9.0 * Celsius)/5.0  +  32;

} /*  Fahrenheit ends*/

/* Definition of PrintAll, module 3000, which prints all results */
void PrintAll(float Fahren1, float Fahren2, float Fahren3, float
Celsius1, float Celsius2, float Celsius3, float Fahren_out1, float
Fahren_out2, float Fahren_out3, float Celsius_out1, float Celsius_out2,
float Celsius_out3)
 {
        printf("The Temperature Conversions from Fahrenheit to Celsius
are\n");
        printf("\t Fahrenheit \t | Celsius \n");
        printf("\t**********************************\n");
        printf("\t %0.2f \t\t | %0.2f \n", Fahren1, Celsius_out1) ;
        printf("\t %0.2f \t | %0.2f \n", Fahren2, Celsius_out2) ;
     printf("\t %0.2f \t | %0.2f \n", Fahren3, Celsius_out3) ;

        printf(" ***** \n ") ;
        printf("The Temperature Conversions from Celsius to Fahrenheit
are\n");
        printf("\t Celsius \t | Fahrenheit  \n");
        printf("\t**********************************\n");
        printf("\t %0.2f \t\t | %0.2f \n", Celsius1, Fahren_out1) ;
        printf("\t %0.2f \t | %0.2f \n", Celsius2, Fahren_out2) ;
        printf("\t %0.2f \t\t | %0.2f \n", Celsius3, Fahren_out3) ;
      printf("\n \n");
        printf("Thank You for Using the UWin_60-140 Temperature
Conversion System!! \n");
        printf("Have a Nice Day and Come Again !!\n");


}

cezeife@charlie:~/fall15/assignmt$ cc cezeife_asn4_prog1.c
cezeife@charlie:~/fall15/assignmt$ ./a.out
Please type the three fahrenheit temperatures to be converted to
Celsius:
```

```
68 -40 100
Please type the three Celsius temperatures to be converted to
Fahrenheit:
0 -40 20
The Temperature Conversions from Fahrenheit to Celsius are
       Fahrenheit       | Celsius
      **********************************
       68.00            | 20.00
       -40.00      | -40.00
       100.00      | 37.78
 *****
 The Temperature Conversions from Celsius to Fahrenheit are
       Celsius     | Fahrenheit
      **********************************
       0.00             | 32.00
       -40.00      | -40.00
       20.00            | 68.00


Thank You for Using the UWin_60-140 Temperature Conversion System!!
Have a Nice Day and Come Again !!
cezeife@charlie:~/fall15/assignmt$ exit
exit

Script done on Wed 30 Sep 2015 02:34:00 PM EDT
```

***

Solution of Program 2

```
Script started on Wed 30 Sep 2015 02:38:39 PM EDT
cezeife@charlie:~/fall15/assignmt$ cat cezeife_asn4_prog2.txt

Step 6: Test and Verification
The execution of the program begins in the main module, where the first
Three input data are read for Fahrenheit as Fahren1 = 68, Fahren2= -40,
Fahren3 = 100.  Then, the second set of input data are read for
Celsius1 = 0, Celsius2 = -40 and Celsius3 = 20. The next instruction
Executed is Celsius( );
This instruction calls the function Calc_Celsius( ) by passing no
parameters.
Thus, control is transferred to this function Celsius, where its
instructions
will be executed before going back to the main function at the point of
call.
variable Celsius_out1. Inside the function Calc_Celsius,
the following instructions of the function are executed.
Celsius_out1 = 5.0/9.0 (Fahren1- 32.0);
Celsius_out2 = 5.0/9.0 (Fahren2- 32.0);
Celsius_out3 = 5.0/9.0 (Fahren3- 32.0);
Since all these variables are global variables that can be used by any
functions, the values of the three Celsius global variables are
obtained as:
```

```
Celsius_out1 = 5.0/9.0 * (Fahren- 32.0) = 5.0/9.0 * (68 - 32.0) = 20.
Celsius_out2 = 5.0/9.0 * (Fahren- 32.0) = 5.0/9.0 * (-40 - 32.0) = -40.
Celsius_out3= 5.0/9.0 * (Fahren- 32.0) = 5.0/9.0 * (100 - 32) = 37.8 .
Execution control is passed back to the main function to execute the
next instruction calling the function Calc_Fahr( );
Control is now passed over to the function Calc_Fahr, where the
its next three sequence of instructions given above are executed.
Fahren_out1 = (9 * Celsius1 + 5 * 32);
Fahren_out2 = (9 * Celsius2 + 5 * 32);
Fahren_out3 = (9 * Celsius3 + 5 * 32);
Inside the function Calc_Fahr, the instruction are evaluated as:
Fahren_out1 = (9.0 * Celsius)/5.0  +  32 = 9.0 * 0 /5.0 + 32 = 32.
Fahren_out2 = (9.0 * Celsius)/5.0  +  32 = 9.0 * -40/5.0 + 32 = -40.
Fahren_out3 = (9.0 * Celsius)/5.0  +  32 = 9.0 * 20/5.0 + 32 = 68.
The function PrintAll is called without parameters to print all these
global input and output variables as shown in the sample output in the
script file.

END.



cezeife@charlie:~/fall15/assignmt$ cat cezeife_asn4_prog2.c

#include <stdio.h>
 /*
 Problem: Given the conversion rate between degree Celsius (C) and
degree Fa
hrenheit (F) is C = 5.0/9.0 (F- 32.0), use two functions Calc_Celsius
(wh
ich returns the Celsius equivalent of a Fahrenheit temperature) and
Calc_Fahr(
which returns the Fahrenheit equivalent of a Celsius temperature) to
write an al
gorithm and a program, which prints
iii)    the Celsius temperatures of  three Fahrenheit temperatures
entered at th
e keyboard by the user for input data, and
iv)     the Fahrenheit equivalents of three Celsius temperatures
entered at the
keyboard by the user for input data.

The problem solution prints lines of output in a neat tabular form as
shown in t
he sample output, by calling the functions and without using repetition
instruct
ions.

This solution uses global variables for and not parameters
in functions.

*/
#include <stdio.h>
```

```
/* Global Variables                              */
float Fahren1, Fahren2, Fahren3, Celsius1, Celsius2, Celsius3;

 float  Fahren_out1, Fahren_out2, Fahren_out3, Celsius_out1,
Celsius_out2, Celsius_out3;

/* Three function prototypes  using global variables             */
void Calc_Celsius (void);
void Calc_Fahr(void);
void PrintAll(void);

/* This is the main Module 0000 which calls modules Celsius and
Fahrenheit
*/
void main(void)
{
/* Now the sequence of instructions   */
printf("Please type the three fahrenheit temperatures to be converted
to Celsius:\n");
scanf("%f %f %f", &Fahren1, &Fahren2, &Fahren3);
printf("Please type the three Celsius temperatures to be converted to
Fahrenheit:\n");
scanf("%f %f %f", &Celsius1, &Celsius2, &Celsius3);
Calc_Celsius( );
Calc_Fahr( );
PrintAll( );

}

/***** The function definition for function Celsius which accepts a
Temperature in Fahrenheit and prints its equivalent in Calc_Celsius.
This called
module 1000

*/
void Calc_Celsius(void)
{
Celsius_out1 = 5.0/9.0 * (Fahren1- 32.0);
Celsius_out2 = 5.0/9.0  * (Fahren2- 32.0);
Celsius_out3 = 5.0/9.0 * (Fahren3- 32.0);

}

/***** The function definition for function Calc_Fahr which accepts a
   Temperature in Fahrenheit and prints its equivalent in Celsius.
This called
module 2000

*/

void Calc_Fahr( )
{
Fahren_out1 = (9 * Celsius1)/ 5.0 + 32.0;
```

```
Fahren_out2 = (9 * Celsius1)/ 5.0 + 32.0;
Fahren_out3 = (9 * Celsius1)/ 5.0 + 32.0;

} /*  Fahrenheit ends*/

/* Definition of PrintAll, module 3000, which prints all results */
void PrintAll(void)
{
        printf("The Temperature Conversions from Fahrenheit to Celsius
are\n");
        printf("\t Fahrenheit \t | Celsius \n");
        printf("\t**********************************\n");
        printf("\t %0.2f \t\t | %0.2f \n", Fahren1, Celsius_out1) ;
        printf("\t %0.2f \t | %0.2f \n", Fahren2, Celsius_out2) ;
     printf("\t %0.2f \t | %0.2f \n", Fahren3, Celsius_out3) ;

        printf(" ***** \n ") ;
        printf("The Temperature Conversions from Celsius to Fahrenheit
are\n");
        printf("\t Celsius \t | Fahrenheit  \n");
        printf("\t**********************************\n");
        printf("\t %0.2f \t\t | %0.2f \n", Celsius1, Fahren_out1) ;
        printf("\t %0.2f \t | %0.2f \n", Celsius2, Fahren_out2) ;
        printf("\t %0.2f \t\t | %0.2f \n", Celsius3, Fahren_out3) ;

      printf("\n \n");
        printf("Thank You for Using the UWin_60-140 Temperature
Conversion System!! \n");
        printf("Have a Nice Day and Come Again !!\n");


}


cezeife@charlie:~/fall15/assignmt$ cc cezeife_asn4_prog2.c
cezeife@charlie:~/fall15/assignmt$ ./a.out
Please type the three fahrenheit temperatures to be converted to
Celsius:
68 -40 100
Please type the three Celsius temperatures to be converted to
Fahrenheit:
0 -40 20
The Temperature Conversions from Fahrenheit to Celsius are
       Fahrenheit       | Celsius
      **********************************
       68.00            | 20.00
       -40.00     | -40.00
       100.00     | 37.78
 *****
 The Temperature Conversions from Celsius to Fahrenheit are
       Celsius     | Fahrenheit
      **********************************
       0.00             | 32.00
       -40.00     | 32.00
```

```
       20.00            | 32.00


Thank You for Using the UWin_60-140 Temperature Conversion System!!
Have a Nice Day and Come Again !!
cezeife@charlie:~/fall15/assignmt$ exit
exit

Script done on Wed 30 Sep 2015 02:40:35 PM EDT
```