

**60-140-1 and 60-140-2 ASSIGNMENT #3 SOLUTION**  
**Handed Out: Thurs. Oct 8, 2015 for (60-140-01 and 60-140-02)**  
**Due: Thurs Oct 22, 2015 for (60-140-01 and 60-140-02)**  
**Total: 50 marks**

---

**Objective of Assignment:** To write an algorithm and a C program to solve a problem using functions and function calls but no decision or repetition instructions by going through the 6 problem solving steps.

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

**Other Things to learn from Assignment :** How to make function calls and use of **call-by-value parameters**.

DO NOT USE GLOBAL VARIABLES

**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 #3 (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 #3 (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.

---

**CONFIDENTIALITY AGREEMENT & STATEMENT OF HONESTY**

**I confirm that I will keep the content of this assignment/examination confidential.**

**I confirm that I have not received any unauthorized assistance in preparing for or doing this assignment/examination. I confirm knowing that a mark of 0 may be assigned for copied work.**

\_\_\_\_\_  
Student Signature

\_\_\_\_\_  
Student Name (please print)

\_\_\_\_\_  
Student I.D. Number

\_\_\_\_\_  
Date

---

**Problem:**

You are required to write an **algorithm and a C program using top-down design**

**approach with functions and only Call-by-Value parameters**, and showing all 6 problem solving steps to compute and print the following:  
 Given the conversion rate between degree Celsius ( $^{\circ}\text{C}$ ) and degree Fahrenheit ( $^{\circ}\text{F}$ ) is  $^{\circ}\text{C} = 5.0/9.0 (^{\circ}\text{F} - 32.0)$ , 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 an algorithm and a program, which prints

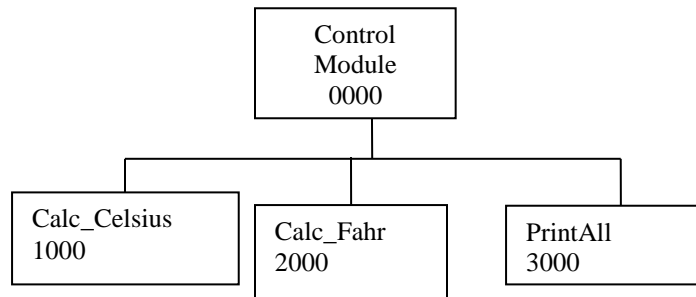
- (i) the Celsius temperatures of three Fahrenheit temperatures entered at the keyboard by the user as input data, and
- (ii) the Fahrenheit equivalents of three Celsius temperatures entered at the keyboard by the user as input data.

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.

All input data should be read in the control or main module.  
 Calc\_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.

All input data must be read in the control or main module and all results printed by another function dedicated to printing. **Do not use GLOBAL variables and use only call-by-value parameters and the following structure chart for solving the problem.**

\*\*\*\*\*



**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 all the 6 problem solving steps for solving this problem in a text file called your userid\_asn3.txt. Then, type the C program solution into a source file called your userid\_asn3.c. Compile your C program with cc userid\_asn3.c. Then, run (execute it with ./a.out). When your program is running with no errors, then, create a script file called userid\_script3.txt and hand in your script file for marking. Your script file should show all 6 problem solving steps including the algorithm, your source program, the compilation of the program, the running of the program with the input and output data shown. Thus, to get your script file you can do the following:

```

script userid_script3.txt
cat userid_asn3.txt
cat userid_asn3.c
cc userid_asn3.c
./a.out
(when prompted, type in the input data in the correct order)
exit

```

Once you exit, your script file (userid\_script3.txt) has been created and you can hand in your script file for marking.

**Marking Scheme :**

1. Correct Problem Analysis in Steps 1 and 2 of problem solving ---(5 marks)
2. Correct Algorithm solution in Step 4 -----(5 marks)
3. Correct Program Solution (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)
4. Hand tracing the program for testing and evaluation(step 6) ---(5 marks)
5. Error free Compilation of Program (syntax handling) --- (5 marks)
6. Correct Running of the Program with Correct Input/Output Data Format ----(10 marks)

---

## PROBLEM SOLUTION

---

The steps for solving the problem including the algorithm, the compilation of the program, and its execution with sample input data as well as hand tracing of the algorithm and program are given in a script file with content shown below.

```
Script started on Wed 30 Sep 2015 12:48:19 PM EDT
cezeife@alpha:~/fall15/assignmt$ cat cezeife_asn3.txt
Step 1 : 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 an algorithm and a program, which prints
iii) the Celsius temperatures of three Fahrenheit temperatures
entered at
the 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 the sample output, by calling the functions and without using repetition instructions.

Step 2 : Problem Analysis

Input Data:

Real Fahren1, Fahren2, Fahren3, Celsius1, Celsius2, Celsius3;

Output Data:

Real Fahren\_out1, Fahren\_out2, Fahren\_out3, Celsius\_out1, Celsius\_out2, Celsius\_out3;

Other Variables:

Relationships:

```
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);
Fahren_out1 = (9 * Celsius1)/ 5.0 + 32.0;
Fahren_out2 = (9 * Celsius1)/ 5.0 + 32.0;
Fahren_out3 = (9 * Celsius1)/ 5.0 + 32.0;
```

Step 3 :

The structure chart for the top-down design is already given. The program must have three sub-modules(functions)  
- one (Calc\_Celsius) for calculating the Celsius of every given Fahrenheit input and the other (Calc\_Fahr) for computing the Fahrenheit of every given Celsius input data. The PrintAll function is used to print the results.

Step 4 : (Algorithm)

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

Module 0000

MainModule( )

{

Input Data:

Real Fahren1, Fahren2, Fahren3, Celsius1, Celsius2, Celsius3;

Output Data:

Real Fahren\_out1, Fahren\_out2, Fahren\_out3, Celsius\_out1,  
Celsius\_out2, Celsius\_out3;

/\* Now the sequence of instructions \*/

Read (Fahren1, Fahren2, Fahren3, Celsius1, Celsius2, Celsius3);

Celsius\_out1 = Calc\_Celsius(Fahren1);

Celsius\_out2 = Calc\_Celsius(Fahren2);

Celsius\_out3 = Calc\_Celsius(Fahren3);

Fahren\_out1 = Calc\_Fahr(Celsius1);

Fahren\_out2 = Calc\_Fahr(Celsius2);

Fahren\_out3 = Calc\_Fahr(Celsius3);

PrintAll(Fahren1, Fahren2, Fahren3, Celsius1, Celsius2, Celsius3,

Fahren\_out1, Fahren\_out2, Fahren\_out3, Celsius\_out1,

Celsius\_out2, Celsius\_out3);

}

1000

Real Calc\_Celsius(Real Fahren)

{

Real Celsius;

Celsius = 5.0/9.0 \*(Fahren- 32.0);

Return (Celsius);

}

2000

Real Calc\_Fahr(Real Celsius)

{

Real Fahren;

Fahren = (9.0 \* Celsius)/5.0 + 32;

Return (Fahren);

} /\* Fahrenheit ends\*/

/\* Definition of PrintAll \*/ 3000

void PrintAll(Real Fahren1, Real Fahren2, Real Fahren3, Real  
Celsius1, Real Celsius2, Real Celsius3, Real Fahren\_out1, Real

```

Fahren_out2, Real Fahren_out3, Real Celsius_out1, Real Celsius_out2,
Real Celsius_out3)
{
    Print("The Temperature Conversion Celsius to Fahrenheit
are\n");
    Print("\t Celsius | Fahrenheit\n");
    Print("\t %0.2f \t | %0.2f \n", Fahren1, Celsius_out1) ;
    Print("%0.2f \t | %0.2f \n", Fahren2, Celsius_out2) ;
    Print("\t %0.2f \t | %0.2f \n", Fahren3, Celsius_out3) ;

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

}

```

Step 5 : Coding

```

#include <stdio.h>
/*
    Problem: Given the conversion rate between degree Celsius (\260C)
and degree Fahrenheit (\260F) is  $\260C = 5.0/9.0 (\260F - 32.0)$ , 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 an
algorithm and a program, which prints
iii) the Celsius temperatures of three Fahrenheit temperatures
entered at the
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 the sample output, by calling the functions and without using repetition instructions.

```

*/
#include <stdio.h>

/* Three function prototypes */
float Calc_Celsius (float);
float Calc_Fahrenheit(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);
Celsius_out1 = Calc_Celsius(Fahren1);
Celsius_out2 = Calc_Celsius(Fahren2);
Celsius_out3 = Calc_Celsius(Fahren3);
Fahren_out1 = Calc_Fahr(Celsius1);
Fahren_out2 = Calc_Fahr(Celsius2);
Fahren_out3 = Calc_Fahr(Celsius3);
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 Calc_Celsius which
accepts a
Temperature in Fahrenheit and prints its equivalent in Celsius.
This called
module 1000

*/
float Calc_Celsius(float Fahren)
{
float Cels;
Cels = 5.0/9.0 * (Fahren- 32.0);
return (Cels);
}

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

*/

float Calc_Fahr(float Celsius)
{

```

```

float Fahren;
    Fahren = (9.0 * Celsius)/5.0 + 32;

    return (Fahren);
} /* 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");

}

```

#### 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\_out1 = Calc\_Celsius(Fahren1);

This instruction calls the function Calc\_Celsius(68) by passing the call-by-value parameter 68 to this

function Calc\_Celsius. Inside the function Calc\_Celsius, the formal parameter, Fahren takes the actual parameter

value of 68 for Fahren1. And the instructions of this function

Calc\_Celsius are executed as follows.

$Cels = 5.0/9.0 * (Fahren - 32.0) = 5.0/9.0 * (68 - 32.0) = 20$ . This value is returned to main for the variable



Celsius\_out1 = 20. The function Calc\_Celsius is called again a second with the instruction Celsius\_out2 = Celsius(Fahren2); This passes a new actual parameter of Fahren2 = -40 into Celsius function. In Calc\_Celsius, the instruction Cels = 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 returns Cels = 5.0/9.0 \* (Fahren- 32.0) = 5.0/9.0 \* (100 - 32) = 37.8 to Celsius\_out3 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: Fahren\_out1 = Calc\_Fahr(Celsius1); Fahren\_out2 = Calc\_Fahr(Celsius2); Fahren\_out3 = Calc\_Fahr(Celsius3); Inside the function Calc\_Fahr, the instruction Fahren = (9.0 \* Celsius)/5.0 + 32; is evaluated each time to return to the main variables the consecutive values Fahren = (9.0 \* Celsius)/5.0 + 32 = 9.0 \* 0 /5.0 + 32 = 32 for Fahren\_out1. Next is Fahren = (9.0 \* Celsius)/5.0 + 32 = 9.0 \* -40/5.0 + 32 = -40 for Fahren\_out2. 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@alpha:~/fall15/assignmt$ cat cezeife_asn3.c
/*
```

```

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 Celsius
(which
returns the Celsius equivalent of a Fahrenheit temperature) and
Fahrenheit
(which returns the Fahrenheit equivalent of a Celsius temperature)
to write
an algorithm and a program, which prints
iii) the Celsius temperatures of three Fahrenheit temperatures
entered at
the 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 the sample output, by calling the functions and without using repetition instructions.

```
*/

#include <stdio.h>

/* Three function prototypes */
float Calc_Celsius (float);
float Calc_Fahr(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);
Celsius_out1 = Calc_Celsius(Fahren1);
Celsius_out2 = Calc_Celsius(Fahren2);
Celsius_out3 = Calc_Celsius(Fahren3);
Fahren_out1 = Calc_Fahr(Celsius1);
Fahren_out2 = Calc_Fahr(Celsius2);
Fahren_out3 = Calc_Fahr(Celsius3);
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 Calc_Celsius which
accepts a
Temperature in Fahrenheit and prints its equivalent in Celsius.
This called module 1000
*/
float Calc_Celsius(float Fahren)
{
```

```

float Cels;
Cels = 5.0/9.0 * (Fahren- 32.0);
return (Cels);
}

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

*/

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

return (Fahren);
} /* 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@alpha:~/fall15/assignmt$ cc cezeife_asn3.c
cezeife@alpha:~/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@alpha:~/fall15/assignmt$ exit
```

```
exit
```

```
Script done on Wed 30 Sep 2015 12:50:43 PM EDT
```