

EEE101 C Programming and Software Engineering

Solutions to Lab Practice 3

Exercise 1

!(a>b)	!(a<=b&& c<=d)	!(a+1==b+1)	!(a<1 b<2&&c<3)
a<=b	a>b c>d	a!=b	a>=b&&(b>=2 c>=3)

Exercise 2

Expression	Parenthesize () Expression	Value
a>b && c<b	(a>b) c<b)	False
a<!b !!a	(a<!b) (!!a)	True
a+b<!c+c	(a+b)<(!c+c)	False
a-x b*c && b/a	(a-x) ((b*c)&&(b/a))	True

Exercise 3

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

```
int main(){
char c='A';
int i=5, j=10;
```

```
printf("%d %d %d\n", !c, !!c, !!!c);          /* This prints: _0 1 0 */
printf("%d %d %d\n", -!i, !-i, !-i-!j);       /* This prints: _0 0 0 */
printf("%d %d %d\n", !(6*j+i-c), !i-5, !j-10); /* This prints: _1 _5 _10 */
```

```
return 0;
}
```

Exercise 4

1. number>= 1 && number<9
 2. ch!='q' || ch !='k'
 3. number>=1&&number<=9&& number!=5
 4. !(number>=1&&number<=9)
-

Exercise 5

Program 1

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

```
int main(){  
int i=7, j=2;
```

```
if (i==1)  
    if (j==2)  
        printf("%d\n",i+i+j);
```

```
printf("%d\n",i=i-j);          /* gets printed 5, because the if (i==1) doesn't hold */  
printf("%d\n",i);             /* gets printed 5 */
```

```
return 0;  
}
```

Program 2

1. When n=100 it prints exact.
2. When n=10 it prints too far.

Exercise 6

The mathematical operation $\min(x,y,z)$ can be represented by the conditional expression:

$$\min(x,y,z) = (x < y) ? ((x < z) ? x : z) : ((y < z) ? y : z)$$

Notes

if $(x < y)$

then $\min(x,z)$ and y will be ruled out

else $\min(y,z)$ and x will be ruled out

Exercise 7

1. Problem statement:

- to ask user to input a real number representing the depth (in kilo- meters) and then compute and display the temperature at the given depth in degree Celsius and Fahrenheit respectively (according to two provided conversion formulas).

2. Analysis:

- inputs:
 - a real number representing the depth (in kilometers).

- outputs:
 - to print/display the temperature at the given depth in degree Celsius and Fahrenheit (printing messages).
- additional requirements or constraints:
 - none (assuming that the user only input valid data, i.e. only real numbers as inputs)

3. Design:

- algorithm
 - 1. declare three variables of the type of float namely depth, Celsius and Fahrenheit, intuitively:
 - * depth - represents the input depth (in kilometers)
 - * Celsius - represents degree Celsius
 - * Fahrenheit - represents degree Fahrenheit
 - 2. ask user to input a real number representing the depth
 - 3. read the input real number and store the value of such a real number into depth
 - 4. compute the temperature at the given depth in degree Celsius according to the provided formula
 - 5. compute the temperature at the given depth in degree Fahrenheit according to the provided formula
 - 6. display the Celsius temperature at the given depth
 - 7. display the Fahrenheit temperature at the given depth

4. Implementation: see the C code with comments.

5. Testing:

- the C program was tested by carrying out a set of experiments; and the C program output was verified successfully. For instance, Please enter the depth (in kilometers) inside the earth as input data.

100

Celsius temperature at depth 100.000000 in km is 1020.000000. Fahrenheit temperature at depth 100.000000 in km is 1868.000000.

Please enter the depth (in kilometers) inside the earth as input data.

250

Celsius temperature at depth 250.000000 in km is 2520.000000. Fahrenheit temperature at depth 250.000000 in km is 4568.000000.

Please enter the depth (in kilometers) inside the earth as input data.
5000
Celsius temperature at depth 5000.000000 in km is 50020.000000.
Fahrenheit temperature at depth 5000.000000 in km is 90068.000000.

```
/*
Name: Simple Program for Depth, Degree Celsius and Degree Fahrenheit Conversion
Copyright: Free
Author: Anonymous Author
Description: Compute and display the temperature at a given depth
inside the earth (as input data) in degree Celsius and
degree Fahrenheit
*/

#include <stdio.h>          /* include information about standard library stdio.h */

main(){                    /* define a function named main */

    float depth, Celsius, Fahrenheit;    /* declare three variables of the type of float
                                           namely depth, Celsius and Fahrenheit */

    printf("Please enter the depth (in kilometers) inside the earth as inputdata.\n");
                                           /* ask user to input a real number representing the
                                           variable depth (in kilometers) */

    scanf("%f", &depth);    /* store the value of such a real number into variable
                               depth depth using thescanf function */

    Celsius = 10*depth + 20;    /* compute the temperature at the given depth in degree
                                Celsius according to the provided formula */

    Fahrenheit = 1.8*Celsius + 32;    /* compute the temperature at the given depth in
                                       degree Fahrenheit according to the provided formula */

    printf("Celsius temperature at depth %f in km is %f.\n",depth, Celsius);
                                           /* display the Celsius temperature atthe given depth */

    printf("Fahrenheit temperature at depth %f in km is %f.\n",depth, Fahrenheit);
                                           /* display the Fahrenheit temperature at the given depth */
}
```

Notes

The solution of this exercise demonstrates solving practical problems in C using the Software Development Method. Students are expected to present their works using the same approach as presented above.

Exercise 8

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

```
#include <math.h>
```

/* math.h is a standard C header file which consists of/declares a set of functions to compute common mathematical operation */

```
main(){
```

```
float number, sqroot;
```

```
printf("Please enter a non negative real number.\n");
```

```
scanf("%f", &number);
```

```
if (number > 0){
```

```
sqroot= sqrt(number);
```

/* sqrt computes square root */

```
printf("Give a non negative real number: %f.\n", number);
```

```
printf("Its square root is: %f.\n", sqroot);
```

```
}
```

```
else{
```

```
printf("Give a non negative real number: %f.\n", number);
```

```
printf("Please provide a non negative one!");
```

```
}
```

```
}
```