ME-172
Computer Programming Language Sessional
Assignment No. 2

Abu Sayeed Roni, ID: 1710065
Dept: ME, Section: B-2

March 2, 2020

Problem 1

Write a C program to find the smallest of 3 integers taken as input using nested if-else statement.

Solution:

The code below could be written a little more succintly if we're allowed to use logical operators and if-else-if chaining. Or if the problem would've asked to find the smallest of, say, 100 integers, we would've definitely used a loop to iterate over the array of integers keeping track of the current smallest integer. But as the problem states that we are to use *nested if-else* statement, here is the code.

```
/**
 * Finds the smallest of three integers.
 */


#include <stdio.h>


int main(void)
{
    // Take user input for three integers.
    int a, b, c;
    printf("%s\n", "Enter three integers:");
    scanf("%i %i %i", &a, &b, &c);

    int smallest;

    // Compare three integers and find the smallest one.
    if (a < b)
    {
        if (a < c)
        {
            smallest = a;
        }
```

```
            else
            {
                smallest = c;
            }
        }
        else
        {
            if (b < c)
            {
                smallest = b;
            }
            else
            {
                smallest = c;
            }
        }

        // Show output.
        printf("Smallest integer: %i\n", smallest);
    }
```

Saving the file with name `smallest.c`. Then Compiling the source file with the following command:

```
$ clang -o smallest smallest.c
```

An executable binary file with name `smallest` will be created if everything goes right.

Output

Running the executable with the following command:

```
$ ./smallest
Enter three integers:
6
3
9
Smallest integer: 3
```

Problem 2

Write a C program to find that whether a year is leap or not.

Solution:

A leap year is a year, occurring once every four years, which has 366 days including 29 February as an intercalary day. But then there is a catch: it cannot be a multiple of 100 because of some special rules that apply every 100 years. But if the year is a multiple of 400, it is a leap year.

```
/**
 * Checks if a given year is a leap year or not.
 */


#include <stdio.h>
#include <stdbool.h>


int main(void)
{
    // Take input from user for a year.
    int year;
    printf("%s", "Enter a year: ");
    scanf("%i", &year);

    // Check if the year is a leap year or not.
    bool leapYear;
    if (year % 400 == 0 || (year % 100 != 0 && year % 4 == 0))
    {
        leapYear = true;
    }
    else
    {
        leapYear = false;
    }
```

```
        // Show output.
        if (leapYear)
        {
            printf("%i is a leap year.\n", year);
        }
        else
        {
            printf("%i is not a leap year.\n", year);
        }
    }
```

Saving the file with name `leapyear.c`. Then compiling the source file with the following command:

```
$ clang -o leapyear leapyear.c
```

An executable binary file with name `leapyear` will be created if everything goes right.

Output

Running the executable in the following manner a few times:

```
$ ./leapyear
Enter a year: 2019
2019 is not a leap year.

$ ./leapyear
Enter a year: 2020
2020 is a leap year.

$ ./leapyear
Enter a year: 2100
2100 is not a leap year.

$ ./leapyear
Enter a year: 2000
2000 is a leap year.
```

NOTE: All the programs are written in *Linux* environment. The executable binary files don't have an extension like .exe, .dmg, or .app beacuse in Linux whether a program is executable or not is determined by the permissions on the file, not the extension. And LLVM's *Clang* is used for the compilation of above source files.