National University of Singapore
School of Continuing and Lifelong Education

TIC1001: Introduction to Computing and Programming I
Semester I, 2020/2021

Problem Set 1
Basic C/C++ Expressions

Release date: 13th August 2020, 8:00 pm
Due: 30th August 2020, 6:00 pm

Information

In this problem set, you will wet your hands writing C/C++ expressions with some useful functions.

Task 1: Freezer (2 marks)

The temperature of an object placed in a freezer will change according to this function:

$$T = T_0 - (T_0 - T_f) \times (1 - e^{-c \times t})$$

where

- T_0 is the initial temperature of the object
- T_f is the temperature of the freezer
- c is some coefficient of heat transfer
- t is the time in hours

Complete the function `double freezer(int hrs, int mins, int t0, double c)`, that takes in four inputs: hours, minutes, initial temperature and the coefficient. It should return the temperature of the object after it has been placed in a -10 degrees freezer for that given duration.

You may take the value of e to be 2.71828.

Tip: You can use the math function `pow(x, y)` which returns the result of x^y, by adding `#include <math.h>` to the top of your file.

Task 2: Temperature Conversion (2 marks)

The formula to convert between Fahrenheit to Celsius is:

$$T_{^\circ C} = (T_{^\circ F} - 32) \times \frac{5}{9}$$

Implement the functions `f_to_c` and `c_to_f` which takes in a temperature in degrees Fahrenheit and converts it to degrees Celsius, and vice versa.

Task 3: Time Elapsed (3 marks)

Given any two times of the day, we can represent the number of hours, minutes and seconds that elapsed between the two times as hh:mm:ss. Following standard convention, mm and ss should be less than 60.

The functions `hrs_elapsed`, `mins_elapsed` and `secs_elapsed` returns the value of hh, mm and ss, respectively.

The inputs for all three functions are six integers, representing the starting hour, minute and second, and the ending hour, minute and second, in 24-hour clock format.

For example, between 12:20:30 pm and 1:30:50 pm, there is a elapsed time of 1 hr, 10 mins and 20 secs.

Hence after running these statements:

```
int h = hrs_elapsed(12, 20, 30, 13, 30, 50);
int m = mins_elapsed(12, 20, 30, 13, 30, 50);
int s = secs_elapsed(12, 20, 30, 13, 30, 50);
```

The value of the variables `h`, `m` and `s` would be 1, 10, and 20, respectively.

You may assume that the start and end times are within the same day and that the starting time is **not later** than the ending time.

Implement the three functions `hrs_elapsed`, `mins_elapsed` and `secs_elapsed`.

Task 4: IP Address (3 marks)

An IP Address is a numerical identification assigned to each device in a network. It is usually displayed in a human readable form consisting of 4 numbers separated by a decimal, e.g. 192.168.0.1.

Each number is actually a representation of an 8-bit binary number. A binary number only contains the digit 0 and 1, and each digit place is referred to as a bit. Thus, the range for an 8-bit binary number is from 00000000 to 11111111.

To convert a binary number to decimal, note that both system a positional numerical systems. In a binary system, the first 8 positions of the digits represents:

$$2^7, 2^6, 2^5, 2^4, 2^3, 2^2, 2^1, 2^0$$

For example, the binary number 11001001 is calculated to be:

2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0	
1	1	0	0	1	0	0	1	
128	64	0	0	8	0	0	1	$= 201$

Implement the function `ip_octet` which takes an 8-digit binary number as input and returns the respective decimal number.

Note: In C, any number that begins with a `0` will be treated as a base-8 number. So remove all leading zeros from your integer when testing, e.g. for `00110011`, just type `110011`.