

COMP 2160 Programming Practices

Lab 6

Nov. 21th - 23th

Objectives

- Space Optimization
- Bit Manipulation (hint, you need it for the solution)

Lab Requirements

Column 10 of *Programming Pearls* (Column 9 in the first edition) must be read prior to your lab. Just like code optimization, you have to take things with a grain of salt. Most often what you have is good enough but occasionally you have to do better (without obfuscation of course!).

Exercise 1

Do problem 8 of Column 10 for dates. If you have extra time you can look at names (think about how you could use the technique for dates on characters).

To do this we will pack 2 decimal digits into a single byte. You can use integer arithmetic (division and modulus) to extract the individual decimal digits, use a left shift operation to "move" the current digits to the left, then add the next digit into the result.

You are actually making use of Binary Coded Decimal (BCD). BCD uses 4 bits to encode the 10 decimal digits. If you look at hexadecimal values you will notice that we can use 4 bits to represent a number between 0 and 9. Therefore you can convert any number into a sequence of hex digits that read just like the original number:

3594 would be converted to 0x3594.

Write a program that reads in a date (from standard input), converts it to BCD as described above, and prints the original date and its BCD equivalent (in hexadecimal format). The program should continue to execute until EOF is detected. For dates, assume that you will always enter them like:

13-7-2003 (i.e. day, a month, and a 4 digit year).

This implies reading the date into 3 variables and then putting it into a single variable in packed BCD format. Don't worry about validating the day value relative to the month it's in or with ensuring that you have the correct number of digits for each value (this would be required if done for real!).

Your output should look something like:

13-7-2003 13072003

Notice that an 8-character date can be stored in 4 bytes (which gives you the data type required...). Skeleton code is provided for this lab.