CPSC 121: Models of Computation
Assignment #2, due Friday October 4th, 2013 at 16:00

[8] 1. Computers represent characters by associating with each character a specific sequence of 0's and 1's. In this question, you will be dealing with the ASCII and Unicode encodings discussed in class. You can find a table listing the decimal (labeled Dec) and hexadecimal (labeled Hx) codes for each ASCII character (labeled Chr) at

http://www.lookuptables.com

by clicking on the "ASCII table" button near the top of the page. This site also contains tables for Unicode accessible by clicking on the "Unicode v4 Tables" button.

[2] a. Linux and MacOS X systems terminate each line of a text file with a single linefeed (LF) character. Windows terminates each line with two characters: a carriage return character (CR) followed by a linefeed character (LF). What are the ASCII codes for these two characters?

[2] b. In one sentence, explain how given its ASCII code, you would convert an uppercase letter into the corresponding lowercase letter.

[1] c. Is the operation from part (c) simpler to describe in decimal or in binary? Why?

[1] d. What is the hexadecimal Unicode character for a pencil pointing towards the right (hint: see "Dingbats")?

[2] e. In Dr. Racket, ASCII and Unicode characters are represented by `#\` followed by the character. For instance, `#\A` is the character "uppercase A", and `#\space` is the space character. Write a Dr. Racket function `numerical-value` that takes as input a single character (that you can assume represents a digit), and returns the numerical value of that character. For instance,

```
> (numerical-value #\8)
8
```

Hint: use the function `char->integer` described in the Dr. Racket documentation.

[10] 2. Let us consider a system where every integer is unsigned and uses 12 bits. Most programming languages provide ways to manipulate the bits of integers (signed or unsigned). For instance, if x and y are two such integers, then

- $x \,\&\, y$ denotes the integer whose i^{th} bit is the i^{th} bit of x **and** the i^{th} bit of y. For instance, assuming that we have integers $746_{10} = 001011101010_2$ and $1493_{10} = 010111010101_2$, then $746 \,\&\, 1493 = 000011000000_2 = 192_{10}$.
- $x \mid y$ denotes the integer whose i^{th} bit is the i^{th} bit of x **or** the i^{th} bit of y. For instance, $746 \mid 1493 = 011111111111_2 = 2047_{10}$.

- $x \hat{} y$ denotes the integer whose i^{th} bit is the i^{th} bit of x **xor** the i^{th} bit of y. For instance, $746 \hat{} 1493 = 011100111111_2 = 1855_{10}$.
- $x \ggg i$ denotes the integer obtained by "deleting" the rightmost i bits of x, and adding i 0 bits to the left of the remaining bits. For instance, $746 \ggg 4 =$ **0000**00101110 (the bits 1010 were removed).

We can perform several operations simply by manipulating the sequence of bits that represents an integer using these operators. In this problem, we look at some of them.

[2] a. There are several situations where we would like to only retain some of the bits of an integer x. For instance, the last 6 bits of 746 equal $101010_2 = 42_{10}$. Given x, what expression would give you an integer that corresponds to the last 6 bits of x?

[2] b. Continuing from the previous question, what expression would give an integer that corresponds to the next four bits of x (those corresponding to the positions 2^9, 2^8, 2^7 and 2^6)? For instance, for 746, this would be $1011_2 = 11_{10}$.

[6] c. In the UNIX file system, there are three types of access permission:

- **read**: determines if someone can open a file for reading, or list files in a directory (folder).
- **write**: determines if someone can update a file, or create new files in a directory (folder).
- **execute**: determines if someone can execute a file (program) or access files inside a directory (folder).

Each can be either allowed or disallowed. These permissions are stored in groups of three: there is one permission for the owner of the file or directory, one for the group the file belongs to (a group could be something like "faculty", or "ugrad-student"), and finally one for everybody else.

For instance, the pattern 111 101 100 means, in order:

- The file owner can read, write or execute the file.
- Other users in the file's group can read or execute it, but not write to it.
- Users other than the owner or members of the file's group can only read it.

Suppose that the permissions for a file are stored as a 12-bit integer (i.e. the permissions shown in the previous paragraph will be stored as $000111101100_2 = 492_{10}$). To set or change permissions, the system performs bit-wise operations on the permissions and predefined constant values. If p is the number representing the permissions of a file, write one or more expressions that will set the permissions p in the way described in each of the following cases. If you use constants you can write them in either decimal or in binary notation:

i. Allow the owner and members of the file's group to read, write or execute it, but other users to only execute it.

ii. Give everyone read access to the file, leaving all other permissions unchanged.

iii. Remove write permissions to the file from everyone except its owner, leaving all other permissions unchanged.

[14] 3. Some early computers supported an integer encoding different from the one discussed in the readings and in class. "Binary Coded Decimal" (BCD) represented d decimal digits using $4d$ bits in groups of 4. Each group of 4 represented a single digit (from 0 to 9). For instance, 78 would be represented as a 7 (0111) followed by a 8 (1000), that is as 01111000.

[3] a. What unsigned decimal integer is represented by the binary value 10010010 if it is interpreted (i) as a binary integer using our usual representation? (ii) as a binary coded decimal integer?

[2] b. There are six "wasted" bit patterns when we represent a 1-digit integer with BCD (the six bit patterns 1010 to 1111 have no meaning in the representation). How many wasted bit patterns are there when we represent a 2-digit integer in BCD (hint: it's not 36)?

[3] c. Find a formula for the number of "wasted" bit patterns when we represent a d-digit decimal integer where $d \geq 1$. Hint: there is a short answer, so if you start doing lots of computations you are on the wrong track.

[2] d. Can we use the "wasted" bit patterns in a 1-digit BCD integer to encode whether the number is positive or negative, allowing us to represent the integers between -9 and 9 instead of just from 0 to 9? Explain why or why not.

[2] e. Can we use the "wasted" bit patterns in a 2-digit BCD integer to encode whether the number is positive or negative, allowing us to represent the integers between -99 and 99 instead of just from 0 to 99? Explain why or why not.

[2] f. Give one advantage of BCD integer representation compared with the representation discussed in the readings and in class.

[8] 4. Determine the validity of the following argument *using rules of inference*. You may use inference rules from class or one of the texts (Epp or Rosen), or you may use a logical equivalence from class or the texts as an inference rule. You must state what

inference rule or logical equivalence you use at each step.

1. $q \rightarrow (s \vee r)$
2. $(\sim p \vee t) \rightarrow q$
3. $\sim u \rightarrow \sim r$
4. $\sim s \wedge \sim u$

$\therefore \sim t$

[10] 5. A system administrator for *ProfileBook* is trying to determine how all of its users most private information ended up publically available on the internet. He gathers the following information:

- Either Bjerc Squale, who graduated from Middle-Earth University, has taken a software-engineering course, or user-provided input is used without validation.
- Either Fidei Phat did not review the code carefully, or commands can not be inserted into SQL statements by the user.
- If Middle-Earth University is not a good university, then its graduates have not taken a software-engineering course.
- If commands can not be inserted into SQL statements by the user, then user-provided input is validated before being used.
- Middle-Earth University is not a good university.

[3] a. Rewrite each of these statements using propositional logic. Make sure to define the propositions you are using. That is, state something like: "b: Bjerc Squale has taken a software engineering course" before writing a proposition involving b.

[7] b. Using your answer from part (a), known logical equivalences, and the rules of inference, prove that Fidei Phat did not review the code carefully (and should be fired).