CPSC 121 2016W2, Midterm 1 Pre-Reading

Gao, Vogt, and Wolfman

February 2, 2017

Two of the problems on our upcoming midterm begin with a section describing the domains of the problems. We include those domain descriptions here so you can read them now and avoid spending time reading them during the exam. On the exam, we'll include a statement saying that the "text exactly repeats the pre-reading you were given", so that you know which blocks of text you have already seen.

Predicate Logic

Consider the following definitions:

- A: all (non-human) animals near the barn.
- P: all people near the barn.
- L: all locations near the barn.
- *Cow*(x): animal x is a cow.
- *Pig*(x): animal x is a pig.
- *Friends*(x, y): x and y are friends.
- *Noise*(x, y): x has made an animal noise at y.
- *Enjoy*(x, y): x enjoys spending time at location y.

For example, we can translate the English statement "Alice enjoys spending time anywhere near the barn" into predicate logic as $\forall x \in L, \textit{Enjoy}(\text{Alice}, x)$.

Number Representation

In our discussion of number representation, we usually discuss **fixed-width** representations. That is, a number might be represented by 8 bits or 32 bits, but not by "however many bits it takes".

However, when people write out numbers, we usually use a variable-width representation. For instance, the grade you get for this course might be a 3-digit number (100), a 2-digit number (10–99), or (hopefully not) a 1-digit number (0–9). If grades were "fixed-width", then, we'd have to write a grade like 83 as 083 instead. An apparent advantage of our variable-width representation is that we can write numbers as large as we like just by taking up more space.

For this problem, we'll consider an approach that uses bits to achieve this sort of variable-width representation of unsigned binary values. We call our proposed representation "flag bit" numbers: a number is composed of one or more 2-bit "blocks". The first bit in each block is a normal base-2 digit. The second bit is the "flag bit". It is 0 if this is the last block in the number and 1 if there is another block. To determine the value of the variable-width number, we collect up all the normal digits, **reverse them**, and interpret the result as an unsigned binary number.

For clarity, we'll write "flag bit" numbers with commas separating the blocks, although those wouldn't be included in the stored representation of the numbers.

For example, the numbers 0 and 1 would be 00 and 10 (respectively) as "flag bit" numbers.

The numbers 2 and 3 would be 01,10 and 11,10 (respectively).

The number 01,11,01,11,10 is the decimal number 26 because: the first four blocks' flag bits indicate that the number continues up to the fifth block's flag bit, which indicates that the number ends there; the binary digits in those five blocks, read from left to right, are 01011; we reverse these bits to get 11010; and, that unsigned binary number is the decimal number 26.