Problems for Tutorial 7 (9/30/99)

Problem 1. A Boolean function from the naturals to $\{0,1\}$ assigns 0 or 1 to every natural number. A computer program is a finite piece of code (in your favorite programming language). The computer program for a Boolean function takes as input a natural number and outputs 0 or 1.

Prove that not all Boolean functions f are computable (i.e can be programmed) using a finite piece of code). Do so by using countability arguments.

(a) What is the cardinality of the set of finite computer programs? Show that this is a countable set. (you may assume that a computer program is written in ASCII).

We can think of any finite program as a string on a finite sized alphabet (the alphabet is the set of possible instructions in the program). In particular, you may think of this as the ASCII alphabet. Let us order the elements of the alphabet as $a_1, a_2, \ldots, a_k$. Then we order the set of all strings on this alphabet as follows: first according to length (strings of length 1, followed by strings of length 2, .. and so on) and then lexicographically (i.e dictionary order) among strings of the same length.

Given a program P define $f(P)$ as the position of P in lexicographic order. This is a map from the set of programs to the set of natural numbers. It is easy to see that the map is an injection. No two programs are mapped to the same natural number. Also every finite program is mapped to some finite natural number. This is because the position in lexicographic order of a program of length n is at most k^{n+1}.

(b) Show that the cardinality of the set of all Boolean Functions is $|2^{\mathcal{N}}|$.

We will show a bijection between the set of all Boolean functions to the power set of the naturals. Define function G which maps a Boolean function f to a subset of the naturals $S = \{i \in N \text{ s.t. } f(i) = 1\}$. This is an injection as if two functions f_1 and f_2 map to the same subset S, then means that $f_1(i) = f_2(i)$ for all i, and thus $f_1 = f_2$. Moreover, it is a surjection as for each subset S of the natural numbers, there exists some function f such that $f(i) = 1$ exactly when $i \in S$. Thus it is a bijection. Since the cardinality of the power set of the naturals is $|2^{\mathcal{N}}|$, so is the cardinality of the set of all Boolean functions over the naturals.

(c) From the above, can you deduce that there are uncomputable Boolean functions? argue your answer.

We already showed that there is a bijection between the set of all Boolean functions to the power set of the naturals. Therefore, since the power set of the set of natural numbers is uncountable then so is the set of Boolean functions.

We conclude that as there are countably many computer programs and uncountably many Boolean function, there are more Boolean functions than programs (or there is no surjection from the set of computer programs to the set of Boolean functions). Thus, there is some Boolean function which is not computed by any computer program.

Problem 2. Let $\{0,1\}^n$ be the set of all bit strings of length n.

(a) How many functions are there with domain $\{0,1\}^m$ and codomain $\{0,1\}^n$?

There are 2^m elements in the domain. There are 2^n elements in the codomain. For every element in the domain, we thus have 2^n choices for what we can map it to. Therefore, by the product rule, there are $(2^n)^{2^m}$ possible functions with domain $\{0,1\}^m$ and codomain $\{0,1\}^n$.

(b) A value x is called a *fixed point* of a function f if $f(x) = x$. How many functions from $\{0,1\}^n$ to $\{0,1\}^n$ have *no* fixed points?

We have 2^n elements in the domain, and each of them can possibly be mapped to $2^n - 1$ elements in the codomain—to any element but itself. Therefore, there are $(2^n - 1)^{2^n}$ such functions with no fixed points.

(c) How many functions from $\{0,1\}^n$ to $\{0,1\}^n$ have *at least one* fixed point?

There are $(2^n)^{2^n}$ functions, and $(2^n - 1)^{2^n}$ have no fixed points. Therefore, the remaining $(2^n)^{2^n} - (2^n - 1)^{2^n}$ have at least one fixed point.

(d) How many functions from $\{0,1\}^n$ to $\{0,1\}^n$ have *exactly one* fixed point?

Let's fix a particular point for our fixed point. Now let's consider where the remaining $2^n - 1$ points can be mapped. Each one can be mapped to any spot but itself—thus it can be mapped to exactly $2^n - 1$ possible spots. So there are $(2^n - 1)^{2^n - 1}$ functions with exactly this particular fixed point and no other. Now there are 2^n possible choices for the particular fixed point, so the total number of functions with exactly one fixed point is $2^n(2^n - 1)^{2^n - 1}$.