

Due: Saturday 3/12, 4:00 PM
Grace period until Saturday 3/12, 6:00 PM

Sundry

Before you start writing your final homework submission, state briefly how you worked on it. Who else did you work with? List names and email addresses. (In case of homework party, you can just describe the group.)

1 Unions and Intersections

Given:

- A is a countable, non-empty set. For all $i \in A$, S_i is an uncountable set.
- B is an uncountable set. For all $i \in B$, Q_i is a countable set.

For each of the following, decide if the expression is "Always Countable", "Always Uncountable", "Sometimes Countable, Sometimes Uncountable".

For the "Always" cases, prove your claim. For the "Sometimes" case, provide two examples – one where the expression is countable, and one where the expression is uncountable.

- $A \cap B$
- $A \cup B$
- $\bigcup_{i \in A} S_i$
- $\bigcap_{i \in A} S_i$
- $\bigcup_{i \in B} Q_i$
- $\bigcap_{i \in B} Q_i$

2 Count It!

For each of the following collections, determine and briefly explain whether it is finite, countably infinite (like the natural numbers), or uncountably infinite (like the reals):

- (a) The integers which divide 8.
- (b) The integers which 8 divides.
- (c) The functions from \mathbb{N} to \mathbb{N} .
- (d) The set of strings over the English alphabet. (Note that the strings may be arbitrarily long, but each string has finite length. Also the strings need not be real English words.)
- (e) Computer programs that halt. *Hint: How can we represent a computer program?*
- (f) The set of finite-length strings drawn from a countably infinite alphabet, \mathcal{A} .
- (g) The set of infinite-length strings over the English alphabet.

3 Countability Proof Practice

- (a) A disk is a 2D region of the form $\{(x, y) \in \mathbb{R}^2 : (x - x_0)^2 + (y - y_0)^2 \leq r^2\}$, for some $x_0, y_0, r \in \mathbb{R}$, $r > 0$. Say you have a set of disks in \mathbb{R}^2 such that none of the disks overlap. Is this set always countable, or potentially uncountable?
(*Hint: Attempt to relate it to a set that we know is countable, such as $\mathbb{Q} \times \mathbb{Q}$.*)
- (b) A circle is a subset of the plane of the form $\{(x, y) \in \mathbb{R}^2 : (x - x_0)^2 + (y - y_0)^2 = r^2\}$ for some $x_0, y_0, r \in \mathbb{R}$, $r > 0$. Now say you have a set of circles in \mathbb{R}^2 such that none of the circles overlap. Is this set always countable, or potentially uncountable?
(*Hint: The difference between a circle and a disk is that a disk contains all of the points in its interior, whereas a circle does not.*)
- (c) Is the set containing all increasing functions $f : \mathbb{N} \rightarrow \mathbb{N}$ (i.e., if $x \geq y$, then $f(x) \geq f(y)$) countable or uncountable? Prove your answer.
- (d) Is the set containing all decreasing functions $f : \mathbb{N} \rightarrow \mathbb{N}$ (i.e., if $x \geq y$, then $f(x) \leq f(y)$) countable or uncountable? Prove your answer.

4 Unprogrammable Programs

Prove whether the programs described below can exist or not.

- (a) A program $P(F, x, y)$ that returns true if the program F outputs y when given x as input (i.e. $F(x) = y$) and false otherwise.
- (b) A program P that takes two programs F and G as arguments, and returns true if F and G halt on the same set of inputs (or false otherwise).

5 Kolmogorov Complexity

Compressing a bit string x of length n can be interpreted as the task of creating a program of fewer than n bits that returns x . The Kolmogorov complexity of a string $K(x)$ is the length of an optimally-compressed copy of x ; that is, $K(x)$ is the length of shortest program that returns x .

- (a) Explain why the notion of the "smallest positive integer that cannot be defined in under 280 characters" is paradoxical.
- (b) Prove that for any length n , there is at least one string of bits that cannot be compressed to less than n bits.
- (c) Say you have a program K that outputs the Kolmogorov complexity of any input string. Under the assumption that you can use such a program K as a subroutine, design another program P that takes an integer n as input, and outputs the length- n binary string with the highest Kolmogorov complexity. If there is more than one string with the highest complexity, output the one that comes first lexicographically.
- (d) Let's say you compile the program P you just wrote and get an m bit executable, for some $m \in \mathbb{N}$ (i.e. the program P can be represented in m bits). Prove that the program P (and consequently the program K) cannot exist.
(*Hint*: Consider what happens when P is given a very large input n .)