CS 70 Discrete Mathematics and Probability Theory
Summer 2020 Course Notes

DIS 2A

1 Unions and Intersections

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) $A \cap B$, where A is countable, and B is uncountable always countable $x \in A \cap B$, $x \in A$

(b) $A \cup B$, where A is countable, and B is uncountable never countable

$S_i = (i, i+1)$

(c) $\bigcap_{i \in A} S_i$ where A is a countable set of indices and each S_i is an uncountable set.

$S_1 \cap S_2 \cap S_3 \cap \cdots = \{\} = \phi = (1,2) \cap (2,3) \cap (3,4) \cap \cdots$

$\mathbb{R} \cap \mathbb{R} \cap \mathbb{R} \cap \cdots \quad S_i = \mathbb{R}$

2 Counting Cartesian Products

For two sets A and B, define the cartesian product as $A \times B = \{(a,b) : a \in A, b \in B\}$.

(a) Given two countable sets A and B, prove that $A \times B$ is countable.

(b) Given a finite number of countable sets $A_1, A_2, \ldots, A_n$, prove that

$$A_1 \times A_2 \times \cdots \times A_n$$

is countable.

(c) Consider an infinite number of countable sets: $B_1, B_2, \ldots$. Under what condition(s) is $B_1 \times B_2 \times \cdots$ countable? Prove that if this condition is violated, $B_1 \times B_2 \times \cdots$ is uncountable.

3 Hello World!

Determine the computability of the following tasks. If it's not computable, write a reduction or self-reference proof. If it is, write the program.

P ⇝ TestHalt

(a) You want to determine whether a program P on input x prints "Hello World!". Is there a computer program that can perform this task? Justify your answer.

Assume PrintsHW(f, arg) is computable.

```
1 def helper(P, x):
2     run P(x), suppress printing
3     print("Hello World!")

4 def TestHalt(P, x):
5     if (PrintsHW(helper(P, x))):
          return "halts"
6     else:
7         return "loops"
8     :
      :
```

```
def helper(P, x, k):
    str = P(x)
    for i in range(k-1):        # 0, ..., k-2
        str += "\n print("foo")"
    s+="\n print("hello world")"
    return str
```

k = 5

```
P(x)
print("foo")
print("foo")
print("foo")
print("hello world")
```

```
def testHalt(P, x):
    P' = helper(P, x, k)
    if PHWk(P', x, k):
        return "halts"
    else:
        return "loops"
```

(b) You want to determine whether a program P prints "Hello World!" before running the kth line in the program. Is there a computer program that can perform this task? Justify your answer.

(c) You want to determine whether a program P prints "Hello World!" in the first k steps of its execution. Is there a computer program that can perform this task? Justify your answer.