Computability

CPS 001
Mason F. Matthews
October 19th and 21st, 2005

What is Computer Science?

What can a computer do?
What can a computer not do?

Write Your Names
(or just exercise your curiosity)
Types of Problems

- **Tractable**
  - Problems that *can* be solved by a computer in a “reasonable” amount of time.

- **Intractable**
  - Problems that *can’t* be solved by a computer in a “reasonable” amount of time,
  - But *can* be solved eventually.

- **Non-computable**
  - Problems that can *never* be solved by a computer.
Questions?

- What is a “problem?”

- What if I do something dumb, and it takes forever?

- What does “reasonable” mean?

Is there a path from Ann to Bob?
Is there a path from Ann to Bob?

How much oil can flow?

Can you color this map with 4 colors?
Can you color this map with 3 colors?

Can you color this map with 3 colors?

Can you color this map with 3 colors?

Can you solve this puzzle?
Can you solve this puzzle?

What is the hardest problem that you can think of?

Can you solve this puzzle?

- Does “Intractable” mean impossible?

How can we deal with these problems?
How can we deal with these problems?

- Random Numbers
  - Can “expect” to solve some in reasonable time
- Approximation
  - Can guarantee that we’re “close” to the right answer

Non-Computable Problems

- Problems that cannot be solved by a computer ever
If you start this program, will it ever stop running?

```java
public class Client {
    public static void main(String[] args) {
        int i = 1;
        while(i>0) {
            System.out.println("Still Running.");
            i++;
        }
        return;
    }
}
```

Proof of impossibility!
What else can’t we know about our programs?

- Do two programs do the same thing?
- Do programs have any bugs?
- Do programs do what they’re supposed to do?