Why Study Different Paradigms?

**Introduction**

- Use of abstraction
- Scalable
- Reusable
- Less error prone

**Higher-level description of computation**

- **C**
  ```c
  int a[10];
  int s = 0;
  int i;
  for (i = 0; i < 10; i++) {
    s += a[i];
  }
  ```

- **Haskell**
  ```haskell
  sum [] = 0
  sum (x:xs) = x + sum xs
  s = sum a
  ```

- **Prolog**
  ```prolog
  sum([],0).
  sum([X|XS],S) :- sum(XS,N), S is X+N.
  sum(A,S).
  ```

**Learn about different, more powerful programming abstractions**

- Encapsulation of recursion schemes & control structures
- Partial function application
- Infinite data structures

**Inverse computation**

- `s = fold (+) 0 a`