How to handle the loop?

- The goal is to provide a static object as the semantics of a loop
- Recursion gives an operational semantics (good) but is not static (bad)

\[ C[\text{while } B \text{ do } C] = \lambda s. B[B][s] \rightarrow C[\text{while } B \text{ do } C][C[C][s]][s] \]

Unwanted recursion

Solutions to recursive function equations

- This equation has many solutions
  \[ q = \lambda n. n \text{ equals zero } \rightarrow \text{one } q(n \text{ plus one}) \]

- E.g. \[ q = \lambda n. n \text{ equals zero } \rightarrow \text{one } \perp \]
  \[ q(\text{zero}) = \text{one} \]
  \[ q(\text{one}) = q(\text{two}) \]
  \[ q(\text{two}) = q(\text{three}) \]
  \[ \ldots \]
Solutions to recursive function equations

- Additional solutions could be:
  \( q = \lambda n. n \text{ equals zero} \rightarrow \text{one} \rightarrow \text{four} \)
  \( q = \lambda n. n \text{ equals zero} \rightarrow \text{one} \rightarrow \text{twoHundred} \)

- In general there is an infinite family of solutions:
  \( q_k = \lambda n. n \text{ equals zero} \rightarrow \text{one} \rightarrow k \)

The ‘best’ solution?

- Our goals should be:
  * Find at least one solution
  * Find the best solution among many
  * Try to ensure the solution follows operational intuition

- The answer is called fixed point semantics

Factorial

- We will find the best solution to the equation:
  \( \text{fac} = \lambda n. n \text{ equals zero} \rightarrow \text{one} \rightarrow n \text{ times} \left( \text{fac} \left( n \text{ minus one} \right) \right) \)

- Consider the family of successive unfoldings
### Unfolding factorial

- Zero unfoldings is just \( \text{fac} \)
- One unfolding is
  \[ \lambda n. n \text{ equals zero } \rightarrow \text{one} \]
  \[ \land \text{times} \left( \text{fac}(n \text{ minus one}) \right) \]
- Two unfoldings is
  \[ \lambda n. n \text{ equals zero } \rightarrow \text{one} \]
  \[ \land \text{times} \left( (n \text{ minus one}) \text{ equals zero } \rightarrow \text{one} \right) \]
  \[ \land \left( (n \text{ minus one}) \text{ times} \left( \text{fac}(n \text{ minus one}) \text{ minus one} \right) \right) \]

### Th factorial family

- Use a subscript for the number of unfoldings
  \[ \text{fac}_1 = \lambda n. n \text{ equals zero } \rightarrow \text{one} \]
  \[ \land \text{times} \left( \text{fac}(n \text{ minus one}) \right) \]
  \[ \text{fac}_2 = \lambda n. n \text{ equals zero } \rightarrow \text{one} \]
  \[ \land \text{times} \left( (n \text{ minus one}) \text{ equals zero } \rightarrow \text{one} \right) \]
  \[ \land \left( (n \text{ minus one}) \text{ times} \left( \text{fac}(n \text{ minus one}) \text{ minus one} \right) \right) \]
- Etc.

### Factorial behavior

- Write out the (graphs) extensions of each unfolding disallowing the recursive call
  \[ \text{graph} \left( \text{fac}_1 \right) = \left[ \text{zero}, \text{one} \right] \]
  \[ \text{graph} \left( \text{fac}_2 \right) = \left[ \text{zero}, \text{one}, \text{one} \right] \]
  \[ \text{graph} \left( \text{fac}_3 \right) = \left[ \text{zero}, \text{one}, \text{one}, \text{two}, \text{two} \right] \]
  \[ \text{graph} \left( \text{fac}_4 \right) = \left[ \text{zero}, \text{one}, \text{one}, \text{two}, \text{two}, \text{three}, \text{six} \right] \]
  etc.
Observations on factorial

- General pattern
  \( \text{graph}(\text{fac}_{i+1}) = \{ \text{zero, one}, \text{one, one}, \text{two, two}, \ldots [i, i] \} \)
- By inspection
  \( \text{graph}(\text{fac}_i) \subseteq \text{graph}(\text{fac}_{i+1}) \)
- Also
  \( \text{graph}(\text{fac}_i) \subseteq \text{graph}(\text{factorial}) \)
- And then
  \( \bigcup_{j=0}^{i} \text{graph}(\text{fac}_j) \subseteq \text{graph}(\text{factorial}) \)

Reverse reasoning

- If \([a, b]\) is a member of factorial, then it must be a member of some \(\text{fac}_k\).
- This implies
  \( \text{graph}(\text{factorial}) \subseteq \bigcup_{j=0}^{i} \text{graph}(\text{fac}_j) \)
- Therefore we must have
  \( \text{graph}(\text{factorial}) = \bigcup_{j=0}^{i} \text{graph}(\text{fac}_j) \)

Zero unfoldings

- Since
  \( \text{graph}(\text{fac}_0) = \emptyset \)
- We will write
  \( \text{fac}_0 = \lambda n. \bot \)