- Recall the Peano Axioms

- $0 \in Nat$, if $x \in Nat$ then $S(x) \in Nat$, i.e. $Sx \in Nat$.

- $0, S0, SS0, SS0, \ldots$

- alphabet $\{0, S\}$

- $\{0\}$ is the Basis Set for Nat

- Inductive definitions of functions used the set of Natural Numbers as (one set in) the domain of the function.

- For the sequence $5, 8, 11, 14, 17, \ldots$

  1. Base case: $f(1) = 5$

  2. Recurrence Relation: $f(n+1) = f(n) + 3$

- In this case both the domain and codomain is the set of natural numbers $f : Nat \rightarrow Nat$

- It is possible to define sets (and not just functions) inductively.

- The set of natural numbers itself was defined inductively using the Peano Axioms given above.

- Recall our introduction to Logic. We introduced the following symbols:
  - The propositions: $p, q, \ldots$
  - The symbols: (,), ,$/\wedge$, $\vee$, $\Rightarrow$, $\Leftrightarrow$
- Consider these symbols as the letters of an alphabet
- Like other alphabets (e.g. a...z) these letters can be combined to form strings (words) that make sense in some way.
- A language (like English) is made up of words in this way.
- These words can in turn be grouped together according to certain rules (denoted by a grammar) to form grammatically correct sentences.

- Suppose we take a subset (just to make the example easier) of the symbols from logic
- Symbols: Propositions, (,), $\wedge$, $\vee$
- Now we can define a set which we call the set of Simple-Logical Expressions (SL-expressions) as follows:
  1. All propositional variables are SL-expressio
  2. If A is an SL-expression then so is $\neg A$.
  3. If A and B are SL-expressions then so are $(A \wedge B)$ and $(A \vee B)$
  4. Nothing else is an SL-expression
- NOTE: This is an inductive definition using an extension of ordinary induction called *Structural Induction*.

- If the definition on the previous slide is an inductive definition then what are the basis elements of the SL-expressions? (Remember 0 is the basis element for the Natural Numbers)

- In the definition of the Natural Numbers, there was just one rule to specify how to generate the next natural number (by applying the successor function).

- In the example above we have a number of rules to specify how to generate new elements of the SL-expressions

- If p, q, r are propositions then apply these rules and write down some elements of the SL-expressions.

- In the same way that we can use induction to prove properties of the natural numbers we can use structural induction to prove properties of a set defined using structural induction (such as the set of SL-expressions).

- Apply the following approach to prove that a property P holds for the SL-expressions:

  1. Base case for SL-expressions: Prove $P(x)$ where x is a proposition

  2. Assume the inductive hypothesis

  3. Prove: $P(\neg A)$, $P((A \wedge B))$ and $P((A \vee B))$

- Prove the following properties of the set of SL-expressions:

  1. The SL-expression x has as many opening as closing parenthesis

  2. The number of propositional symbols in an SL-expressions is greater than the number of left parenthesis in an SL-expression

- Can you say what the general procedure is for proving a property of a set that is defined using structural induction.