Intro: Today I'm going to introduce a concept called Lyndon words. On their surface, they don't seem particularly relevant to Lie theory or Hopf algebras, but they actually form an algebraically independent quasisymmetric set for a couple different important Hopf algebras. For example, a shuffle algebra over a field of characteristic 0 can be viewed as a polycyclic algebra over the Lyndon words.

Defn. Fix a totally ordered set $A$ called the alphabet.
- A word over $A$ is a finite tuple of elements of $A$. Let $A^*$ denote the set of words over $A$.
- Let $\emptyset$ denote the empty word.
- For a word $w \in A^*$, let $w_i$ denote the $i$th letter of $w$, i.e., the $i$th entry of the tuple $w$.
- Let $|w|$ denote the number of letters in $w$, i.e., the length of the word $w$.
- The concatenation of two words $u,v$ is the word $u_1 \ldots u_{|u|} v_1 \ldots v_{|v|}$, written $uv$.
- A prefix of $w \in A^*$ is a word $w \in A^*$ or $\exists v \in A^*$ s.t. $w = uv$.
2. A suffix or we say barred word \( u \) is a suffix of \( w \) if \( w = uv \). \( V \) is a proper suffix if \( u \neq \emptyset \).

We define a relation \( \leq \) on the set \( \mathcal{A}^+ \) as follows: For \( u, v \in \mathcal{A}^+ \),

- either \( \exists i \in \mathbb{N}_0 : i \leq |u| \) and \( u[i] = v[i] \) for all \( 0 \leq i \leq |u| \), or
- \( u \preceq v \) and \( |u| < |v| \), or \( u = \emptyset \) and \( v \neq \emptyset \), or
- \( u \) is a prefix of \( v \).

Fact: \( \preceq \) totally orders \( \mathcal{A}^+ \). This can be proven with simple case analysis. \( \preceq \) is called the lexicographic order on \( \mathcal{A}^+ \).

Example: \( \preceq \emptyset \leq \emptyset \leq 1 \leq 11 \leq 111 \leq 1111 \leq 11111 \).

Rules: \( \preceq \) does not respect concatenation from the right. For example, \( \emptyset \leq 1 \), \( \emptyset \leq 13 \), \( \emptyset \leq 133 \), \( \emptyset \leq 132 \), \( \emptyset \leq 1321 \), \( \emptyset \leq 13215 \).

Prop 1.2: \( \leq \) on \( \mathcal{A}^+ \).

- a) If \( c \preceq d \), then \( ac \preceq ad \).
- b) If \( c \preceq d \), then \( ac \preceq ad \).
- c) If \( c \preceq d \), then \( ac \preceq ad \).
- d) If \( c \preceq d \), then \( ac \preceq ad \).
- e) If \( c \preceq d \), then \( ac \preceq ad \).
- f) If \( c \preceq d \), then \( ac \preceq ad \).
- g) If \( c \preceq d \), then \( ac \preceq ad \).
- h) If \( c \preceq d \), then \( ac \preceq ad \).
- i) If \( c \preceq d \), then \( ac \preceq ad \).
- j) If \( c \preceq d \), then \( ac \preceq ad \).
- k) If \( c \preceq d \), then \( ac \preceq ad \).
Prop 6.2: Case analysis. Excluded.

Q1: When do works commute?

Prop 6.4: Let \( u, v \in \mathbb{A} \) satisfy \( u \circ v = v \circ u \). Then \( F \circ \varnothing \circ F \varnothing = F \varnothing \circ F \).

Proof: Induction on \( l(u) + l(v) \). Assume \( l(u) < l(v) \).

Observe that either \( u \) or \( v \) is a prefix of \( w \) or vice versa, i.e., the shorter word must be a prefix of the longer one. When \( u = v \), we have \( u = v \).

Then \( F \circ \varnothing \circ F \varnothing = F \varnothing \circ F \).

Also, \( uw = aw \Rightarrow uw = uw \Rightarrow uw = uw \).

By induction, \( F \circ \varnothing \circ F \varnothing = F \varnothing \circ F \).

Then \( F \circ \varnothing \circ F \varnothing = F \varnothing \circ F \), so we're done. \( \square \)

Prop 6.5: Let \( u, w \in \mathbb{A} \) be rewriting rules such that \( u \circ v = v \circ w \). Then \( F \circ \varnothing = F \).

Proof: Induction on \( l(u) + l(v) + l(w) \). This proof is slightly more complicated than the proof of 6.4, but it is not radically different. It uses 6.2 heavily.

For the cases, it is omitted. \( \square \)

Cor 6.6: Let \( u, v \in \mathbb{A} \) be such that \( u \circ v = v \circ u \).

Then \( uv = vu \).

Proof: Assume the contrary, so \( uv > vu \) or \( vu > uv \).

Assume \( uv > vu \) (else, parts A and B). By 6.5,
\[\begin{align*}
\end{align*}\]
Prop 6.14: Let \( W \) be a Lyndon word & \( u \), \( v \) two words such that \( W = uv \).

a) If \( V \) is non-empty, then \( V \geq W \).
b) If \( V \) is non-empty, then \( V \geq u \).
c) If \( u \) or \( V \) is empty, then \( Vu \geq uv \).
d) \( Vu \geq uv \).

Pf: Clear.

Cor 6.15: Let \( W \) be a Lyndon words & \( V \) a non-empty suffix of \( W \). Then \( V \geq W \).

Pf: (6.14).

Prop 6.16: Let \( u, v \) be Lyndon words or \( u \leq v \).

a) The word \( uv \) is Lyndon.

b) We have \( uv \leq v \).

Pf: b) \( u \) is Lyndon so it's non-empty. Then \( uv \leq v \).
Assume \( uv \neq v \). See \( uv \), Prop 6.12) implies \( W = v \), \( u \) is a prefix of \( v \), so \( a \) is a non-empty prefix of \( uv \).
Thus \( I \) is a \( x \) or \( v = u \). Such a \( t \) is non-empty.
else \( u \neq v \). So \( t \) is a non-empty prefix of \( uv \). So \( t \geq v \), Prop 6.12) \( uv \geq u \), see \( uv \neq v \), \( uv \leq v \). This completes Pf a) & b).

a) \( v \neq \emptyset \) or \( v \) is Lyndon. So \( uv \neq \emptyset \). Need to check every non-\( \emptyset \) proper-suffix of \( uv \) \( v \geq u \).
Let \( p \) be non-\( \emptyset \) proper-suffix of \( uv \). There are two cases.
1) \( p \) is a non-\( \emptyset \) suffix of \( v \).

2) \( p \) has \( \emptyset \) suffix \( q \) \( \neq \emptyset \) proper suffix of \( a \).

First, make case 1. See \( v \in \text{Lyndon} \), \( p \neq \emptyset \), \& by prop 12) \( u \leq v \), so \( p \geq v \). \( \& \) \( p \geq v \).

Case 2: \( p = q \), \( q \) is a non-\( \emptyset \) proper suffix of \( a \).

We have \( q > u \). By prop 12), either \( u \neq q \), \& \( u \leq q \) is a prefix of \( q \). See \( u \) is not a prefix of \( q \), \( uv = qv \).

\( uv = qv \) (else \( uv = u \)), we have \( uv \leq qv = p \), \( q \) done.

So \( p \geq uv \), always, so \( u \leq v \) is Lyndon.

Case 6.17: Let \( u \) \& \( v \) be two Lyndon words \( u \leq v \). Let \( z \) be a word \( z \leq v \) \& \( u \leq z \). Then \( z = \emptyset \).

Proof: Assume not. Then by prop 12) \( u \leq zv \). By prop 13) \( u \leq zv \), contradiction.

Proport 6.12: Let \( u \) \& \( v \) be Lyndon, then \( u \leq v \rightarrow uv \leq uu \).

Proof: We have 3 cases:

1) \( u < v \);
2) \( u = v \);
3) \( u > v \).

Case 1: \( u < v \rightarrow uv < v \) (by prop 13) \( u \leq v \)).

Case 2: \( u = v \rightarrow uu = vu \), by prop 12) again.

Case 3: \( u > v \rightarrow vu < u \leq v \), by prop 13) again.

We now define an important feature of Lyndon words:

A \text{Lyndon between all words \& multisets of Lyndon words}.

This is vital for anyony grammar \& its the robust dig.
Theorem 1. Every non-empty regular language is context-free.

Proof. If $L$ is a regular language, then there exists a deterministic finite automaton (DFA) $A$ that recognizes $L$. Let $p$ be the state of $A$ at which $L$ is first recognized. Then, we can construct a pushdown automaton (PDA) $B$ that recognizes $L$ as follows:

1. **Initialization**: $B$ starts in the initial state $p_0$.
2. **Transition**: $B$ follows the transition rules of $A$.
3. **Stack**: $B$ uses the stack to simulate the DFA's transitions. For each transition of $A$, $B$ pops the top symbol of the stack and pushes a new symbol.
4. **Acceptance**: $B$ accepts a string $w$ if it reaches an accepting state on the stack and the string is recognized by $A$.

Thus, every regular language is context-free. Q.E.D.

Example: Consider the language $L = \{a^n b^n | n \geq 0\}$. This language is not context-free, but it is regular. The DFA for $L$ can be constructed as follows:

- **States**: $q_0, q_1, q_2$.
- **Input Symbols**: $a, b$.
- **Transitions**:
  - From $q_0$, on $a$: to $q_1$.
  - From $q_1$, on $b$: to $q_2$.
  - From $q_2$, on $a$ and $b$: to $q_0$.
- **Accepting State**: $q_2$.

This DFA recognizes $L$. However, the corresponding PDA for context-free languages would need an infinite stack to keep track of the $a$'s and $b$'s.
Consider case 1 where \( w \in \mathcal{L}_G \). By Lemma 6.28, there is a sequence \( a_1, a_2, \ldots, a_n \) with \( a_i \in \{a, b, c\} \) for all \( i \), such that \( w = a_1a_2\ldots a_n \). Now, we'll show how to construct a sequence \( b_1, b_2, \ldots, b_m \) of \( b \)'s and \( c \)'s in such a way that the sequence is equal to \( w \).

For each \( a_i \), we want to construct \( b_i \). If \( a_i = c \), then \( b_i = c \). If \( a_i = a \), then \( b_i = a \). If \( a_i = b \), then \( b_i = b \). This construction ensures that the sequence \( b_1b_2\ldots b_m \) is a valid string in \( \mathcal{L}_G \).

Now, consider the case where \( w \not\in \mathcal{L}_G \). We'll prove that \( w \not\in \mathcal{L} \). As before, we construct a sequence \( a_1a_2\ldots a_n \) of \( a \)'s, \( b \)'s, and \( c \)'s. However, we cannot construct a sequence \( b_1b_2\ldots b_m \) such that \( b_i \in \{a, b, c\} \) for all \( i \), for any \( a_1a_2\ldots a_n \).

To prove this, let's assume for contradiction that there exists a sequence \( b_1b_2\ldots b_m \) such that \( b_i \in \{a, b, c\} \) for all \( i \). Let \( k \) be a hypothetical length where \( b_1b_2\ldots b_k \) is a prefix of \( w \). Then, \( a_{k+1} \) must be either \( a \), \( b \), or \( c \). If \( a_{k+1} = a \), then \( b_{k+1} \) must be either \( a \), \( b \), or \( c \). This contradicts the construction of \( b_1b_2\ldots b_m \).

Therefore, we conclude that for any \( a_1a_2\ldots a_n \), there exists at least one \( a_j \) such that \( a_j \not\in \{a, b, c\} \), and hence \( a_1a_2\ldots a_n \not\in \mathcal{L} \).
Now we examine a technique of Lyndon words into smaller words, called subword lexicographic.
It will be useful for proving inclusion over Lyndon words.

**Theorem 6.30**

Let \( w \) be a Lyndon word of length \( n \).

**1.** Let \( v \) be \( n \)-lexicographically smallest renaming properly suffix \( w \). Since \( w \) is proper, \( F(w) \neq \emptyset \) in \( A^n \) or \( w = u \cdot v \). Consider \( u \neq u \).

a) The words \( u \) \& \( v \) are Lyndon.

b) We have \( u \leq w \cdot v \).

**Proof:**

a) Every renaming properly suffix \( v \leq w \) \( \iff \)
Every suffix \( u \cdot v \) is also a suffix of \( w \) \& \( u \cdot v \) was lexically smaller.
Hence \( v \) is proper.
\( v \) is renaming, so \( v \) is Lyndon.

b) Since \( w \) is Lyndon, every NPS \( u \cdot w \) \( \geq \) \( w \).

Here, \( w \cdot v \). \( v \) renaming \( \Rightarrow \) so \( u \cdot w \cdot v \).

**Proposition 6.31**

If \( p \cdot S = \emptyset \), \( p \cdot S \neq \emptyset \) at \( u \), then \( p \cdot v \) \( \in \) NPS \( u \cdot v \) \( \leq \) \( p \cdot w \).

Assume \( T \subseteq A^n \).
Then \( p \cdot S \neq \emptyset \).
Then \( p \cdot S \neq \emptyset \).

We have \( w \cdot v = p \cdot v = p \cdot (w) \) \( \iff \) \( v \) is a proper suffix \( w \).

**Example:** For \( u \neq \emptyset \), \( v \) is a proper suffix at \( w \). Further, \( v \neq \emptyset \), \( v \) is a NPS of \( w \).

By assumption, \( v \leq q \cdot v \).

By prop 6.20, \( p \cdot v \leq p \cdot q \cdot v \).
\[ p \in p^1 u = w, \text{ contradicting } p^1 \succ w. \]

Thus, it is false that \( p \in u, \) so \( p \not\succ u, \) \( p \) was arbitrary, so \( W \) is Lyndon. \( \Box. \)

We now want to connect the theory of Lyndon words with the notion of shuffle products. Recall:

**Defn:** a) Let \( n \in \mathbb{N} \), the Shannon denotes the word:

\[ \{ v \in \text{Shw}_{mn} : v^{-1}(n) = 0, v^{-1}(m) = n \} \]

b) Let \( u = (u_1, \ldots, u_m) \) and \( v = (v_1, \ldots, v_n) \) be two words.

If \( \sigma \in \text{Shw}_{m,n} \), then \( u\#_\sigma v \) will denote the word

\[(u_0, v_0, u_1, v_1, \ldots, u_m, v_n) \]

The concatenation \( u \# v = (u_1, \ldots, u_m, v_1, \ldots, v_n) \).

Note that the multiset of all \( \# \) letters of \( u \#_\sigma v \) is the disjoint union of the multiset of all letters of \( u \) with the multiset of all letters from \( v \). Hence,

\[ k(u \#_\sigma v) = k(u) + k(v) \]

c) Let \( u = (u_1, \ldots, u_m) \) and \( v = (v_1, \ldots, v_n) \) be two words.

The multiset of shuffles of \( u \# v \) is the multiset

\[ \{ (u_0, v_0, \ldots, v_0) : \sigma \in \text{Shw}_{m,n} \} \]

It is denoted \( u \#_\tau v \).
I will state herein result relating shuffle algebras & Lyndon words.

Thm 6.71 (Radford): Assume $A$ is a subset of $K$.

Let $V$ be a free $K$-module with basis $(a^A)$ and $\alpha$ where $\alpha$ is a totally ordered set. Then the shuffle algebra $Sh(V)$ is a polynomial $K$-algebra. An algebraically independent generator set of $Sh(V)$ can be constructed as follows.

For every word $w$ over the alphabet $A$, let us define an element $b_w$ in $Sh(V)$ by $b_w = b_{w_1} b_{w_2} \ldots b_{w_k}$ where $k$ is the length of $w$. [The multiplicative rule is in $\mathbb{T}(V)$]. Let $\mathcal{X}$ denote the set of all Lyndon words over $A$. Then $(b_w)_{w \in \mathcal{X}}$ is an algebraically independent generator set of the $K$-algebra $Sh(V)$.

Ex: elements of $Sh(V)$ written as polynomials in $b_0$.

- $b_1 = b_{1}$ (since $1$ is Lyndon).
- $b_{21} = b_{1} b_{2} b_{1}$ [ is multiplican in $Sh(V)$].
- $b_{11} = \frac{1}{2} b_{1} b_{1} b_{1}$
- $b_{213} = b_{2} b_{1} b_{3} - b_{123} - b_{13}$
- $b_{321}$
Before we prove this theorem, we'll need the following fact, which provides a non-connection between Lyndon words and shuffles:

**Theorem 6.4:** Let \( u \) and \( v \) be two words. Let \((a_1, \ldots, a_p)\) be the CFL factorization of \( u \), and \((b_1, \ldots, b_q)\) be the CFL factorization of \( v \).

1. Let \((c_1, \ldots, c_{p+q})\) be the result of sorting the task \((a_1, \ldots, a_p), (b_1, \ldots, b_q)\) in descending order. Then the lexicographically highest elt of \( uWv \) is \( c_1 \cdots c_{p+q} \) and \((c_1, \ldots, c_{p+q})\) is a CFL factorization of the elt.
2. Let \( W \) denote the set of all Lyndon words. If \( W \) is a Lyndon word and \( z \) is any word, let \( \text{mult}(z, W) \) denote the number of times in the CFL factorization of \( z \) where \( z \) equals \( W \). The multiplication with which the lexicographically highest elt of \( uWv \) is multiplied \( uWv \) appears in \( \text{mult}(z, W) \).

(The product is well-defined b/c almost all words occur.)

3. If \( a_j \geq b_j \) for \( 1 \leq j \leq p \), then \( \text{mult}(z, W) \) is the highest elt of \( uWv \).
4. If \( a_j > b_j \) for \( 1 \leq j \leq p \), then \( \text{mult}(z, W) = 0 \).
5. If \( a_j = b_j \) for \( 1 \leq j \leq p \), then \( \text{mult}(z, W) = 1 \).
6. Assume \( u \) is Lyndon. If \( u \geq b_j \) for \( 1 \leq j \leq p \), then the highest elt of \( vWu \) is \( uWv \).
Ex. Let $u = 23, \bar{u} = 32, \bar{v} = 21$ and $\Delta = 81/2/3/5$.

The CFL factors are $u & v = (23,23,2)$ & $(3,23,2,2,1)$ respectively. In the variation at time

\[ t = \frac{6}{44}, \quad p = 3, \quad q = 5, \quad \text{with} \]

\[ v = (1,1,v) = (3,23,23,23,2,2,2,1). \]

\[ (\Delta t) = (3,23,23,23,2,2,2,1). \]

Thus, the sequence $(23,23,23,23,2,2,2,1)$ appears in $\text{UWV}$

with a multiplicity of $\Delta (\text{mult}_u + \text{mult}_v)$.

All our factors are $1$, so any factors which are not $1$ are those corresponding to Lyddon

words $w$ which appear in both the CFL factors at $u & v$, i.e., for any other factors, at least one of

$\text{mult}_u$ or $\text{mult}_v = 0$; so $\Delta(\text{mult}_u + \text{mult}_v) = 1$.

In our example, the only factors which are not $1$

are those for $w = 23$ & $w = 2$. So:

\[ \Delta (\text{mult}_u + \text{mult}_v) = \left( \text{mult}_u + \text{mult}_v \right) \left( \text{mult}_u + \text{mult}_v \right) \]

\[ \text{mult}_u = \left( \frac{2+1}{2} \right) + 3(1+1) + 3 \]

\[ = 9. \]

In order to prove Them 6.11, we'll need to prove

some stronger statements, for which we need some new

notation.
Definition (a): If \( p, q \in \mathbb{R} \), then \( [p, q]^+ \) denotes \\
\( \mathbb{R}^+ \cup \{q\} \cup \{p\} \cup \{q, p\}. \)

(b) If \( I, J \) are non-\( \emptyset \) intervals of \( \mathbb{R} \), then \( I < J \) \\
iff every \( i \in I \) is strictly less than \( j \).

c) If \( w \) is a word with \( n \) letters, \( I \) a interval of \( \mathbb{R} \) \\
and \( I \subset [0:n]^+ \), then \( w[I] \) denotes the word \\
\( w_{\mu_1} \ldots w_{\mu_n} \) where \( I = [p, q]^+ \), \( q \geq p \). A \\
word obtained from \( w[I] \) is called a \\
factor of \( w \).

d) Let \( x \) be a composition. Define a tuple \\
\( \text{intsys}(x) \) of intervals of \( \mathbb{R} \) as follows: Write \\
\( x = (x_1, \ldots, x_l) \) where \( l = l(x) \). Then, set \\
\( \text{intsys}(x) = (I_1, \ldots, I_l), \) where \\
\[ I_i = \left[ \sum_{k=1}^{i-1} x_k, \frac{x_i + x_{i+1}}{2} \right]^+ \quad \forall i \in \{1, \ldots, l\}. \]

Then \( \text{intsys}(x) \) is an \( l \)-tuple of \( \text{non-}\emptyset \text{ intervals of } \mathbb{R} \). 
Two tuples are called \text{intervalsystem corresponding to } \alpha.

Example: Let \( \alpha = (4, 1, 4, 2, 3) \). Then the intervalsystem 

\( \alpha \) corresponds to \( \alpha \beta: \)

\( \text{intsys}(\alpha) = (\left[0:4\right]^+, \left[4:5\right]^+, \left[5:9\right]^+, \left[9:11\right]^+, \left[11:14\right]^+) \)

\( = (3, 4, 5, 6, 7, 8) \).
b) If $I \cap J$ is disjoint and non-$\emptyset$ intervals of $\mathcal{P}$, then $I \leq J$ or $J \leq I$.

c) Let $x$ be a component. Write $x = (a_1, \ldots, a_k)$ where $a_i \in \mathcal{P}(x)$. Then $x$ is contained in $1$-tuple $(I_1, \ldots, I_k)$ of non-$\emptyset$ intervals of $\mathcal{P}$ by

1) $I_1, I_2, I_C$ form a set partition of $\mathcal{P}(x)$, $n=1, k$;

2) $I_1 \leq \ldots \leq I_k$;

3) $|I_1| = x; \forall i \in 1, \ldots, k^2$.

Pr: Easy.

The following lemmas are consequences of the definition of ell, s, Shmp, & shuffle alg:

Lemma 6.50: Let $I$ be $\emptyset$ & $I \neq \emptyset$. Let $C \neq \emptyset$.

a) If $I$ is an interval of $\mathcal{P}$ or $I \in [0, n\mathcal{P}]$, then $C(I) \in [0, n\mathcal{P}]$ and $C(I) \cap [n\mathcal{P}]$ is non-$\emptyset$.

b) Let $L, K$ be non-$\emptyset$ intervals of $\mathcal{P}$ or $L \in [0, n\mathcal{P}]$ & $K \notin [0, n\mathcal{P}]$ & $K \cap L$ is a non-$\emptyset$ interval. Assume $s^{-1}(K) \cup s^{-1}(L)$ is non-$\emptyset$. Then $FP \neq \emptyset$ or $C(L) \cap [n\mathcal{P}]$ & $s^{-1}(P), s^{-1}(K) \cup s^{-1}(L)$ are non-$\emptyset$ or $s^{-1}(K) < s^{-1}(P) < s^{-1}(L)$.
c) Lemma 6.50(b) reads only if \( K \in \text{finm}^I \) and \( \sigma \in \text{finm}^I \) are replaced by \( K \in \text{finm}^J \) and \( \sigma \in \text{finm}^J \), respectively.

Lemma 6.52. Let \( u \) and \( v \) be two words, \( u = l(u), v = l(v) \). Let \( \sigma \in \text{finm}^I \).

a) If \( I \) is an interval satisfying either \( I \subseteq \text{finm}^I \) or \( I \subseteq \text{finm}^J \) and \( \sigma^{-1}(I) \) is an interval, then
\[
(6.52) \quad (u \cup v)[\sigma^{-1}(I)] = (uv)[I].
\]

b) Assume \( u \cup v \) is the lex upper cell of \( u \cup uv \).

Let \( I \subseteq \text{finm}^I \) and \( J \subseteq \text{finm}^J \) be two von-\( \emptyset \) intervals. Assume \( \sigma^{-1}(I) \) and \( \sigma^{-1}(J) \) are also intervals, and \( \sigma^{-1}(I) \cap \sigma^{-1}(J) \) is an interval as well. Then \( (uv)[I], (uv)[J] \geq (uv)[I] \cup (uv)[J] \).

c) (6.52(b)) yields if \( I \subseteq \text{finm}^I \) and \( J \subseteq \text{finm}^J \) is replaced by \( I \subseteq \text{finm}^J \) and \( J \subseteq \text{finm}^J \).

Exercise.

If the \( op ) \Rightarrow \text{finm} 6.54,

Ex. 6.55, ---