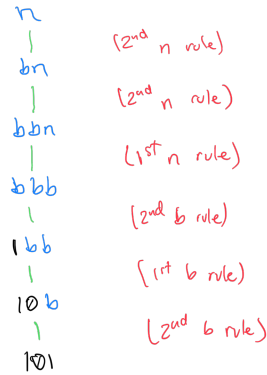


b corresponds to a bit, it can have the value of \emptyset or 1 . No other values are syntactically valid/well-formed.

$n ::= b$
| bn

n corresponds to bit sequences, e.g. $\emptyset, 1, 10, 11, \dots$

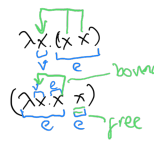
BNF grammar can be used to generate valid expressions or to parse expressions, e.g. the sequence 101 can be generated as an n , as follows:



λ -Calculus Syntax

$v ::= a|b|c|\dots$

$e ::= v$
| $\lambda v.e$
| $(e e)$



v and e are non-terminals corresponding to variables and expressions.

$\lambda, (,)$, are terminal symbols.

$::=, |$ are BNF meta-syntax symbols.

~~λx~~
 ~~$\lambda x.y$~~
 ~~(x)~~
 ~~$(x y \lambda z.w)$~~

$(\lambda x.(x x) \lambda x.(x x))$ ✓
 $(f (g (h x)))$ ✓
 $(x x)$ ✓

$e \xrightarrow{\beta} e' \xrightarrow{\eta} e''$