Grammar

```
A → B impl B
  | B .
B → C mult B
  | C .
C → C arroba
  | D .
D → int
  | float
  | piz A pdr .
```

Some sentences generated by this grammar: {int, float, int arroba, int mult int, int impl int, float arroba, float mult int, int impl float, int mult float, float impl int, float mult float, float impl float, int arroba arroba, float arroba arroba, int mult int arroba, int mult float arroba, int arroba arroba arroba, float arroba arroba arroba, int arroba arroba arroba arroba, float arroba arroba arroba arroba}

- All nonterminals are reachable and realizable.
- There are no nullable nonterminals.
- The endable nonterminals are: D C A B.
- No cycles.

nonterminal	first set	follow set	nullable	endable
A	int float piz	pdr	no	yes
B	int float piz	pdr impl	no	yes
C	int float piz	pdr mult arroba impl	no	yes
D	int float piz	pdr mult arroba impl	no	yes

The grammar is not LL(1) because:

- A has a first set conflict.
- B has a first set conflict.
- C is left recursive.

- attempt to transform the grammar (to LL(1))
- generate LL(1) parsing table
- generate LR(0)/SLR(1) automaton
- generate LALR(1) automaton
- generate LR(1) automaton

Return home to enter a new grammar.