### E+T Grammar

1. <E> → <E> + <T>
2. <E> → <T>
3. <T> → <T> \* <F>
4. <T> → <F>
5. <F> → <P> ↑ <F>
6. <F> → <P>
7. <P> → ( <E> )
8. <P> → CONST

### Left-Factoring

1. <E> → <E> + <T>
2. <E> → <T>
3. <T> → <T> \* <F>
4. <T> → <F>
5. <F> → <P> <POWER-PART>
6. <POWER PART> → ↑ <F>
7. <POWER PART> → ε
8. <P> → ( <E> )
9. <P> → CONST

### Removal of Left Recursion

1. <E> → <T> <E-LIST>
2. <E-LIST> → + <T> <E-LIST>
3. <E-LIST> → ε
4. <T> → <F> <T-LIST>
5. <T-LIST> → \* <F> <T-LIST>
6. <T-LIST> → ε
7. <F> → <P> <POWER PART>
8. <POWER PART> → ↑ <F>
9. <POWER PART> → ε
10. <P> → ( <E> )
11. <P> → CONST

We end up with the <E-LIST> grammar which is LL(1).