```
open      =  symbol '('
close     =  symbol ')'
plus      =  symbol '+'
minus     =  symbol '−'
```

```
data Tree
= Leaf  Int
 | Node Tree Op Tree
type Op = Char
```

```
expr, term :: Parser Char Tree
expr =  Node    ® term ⊗ (plus⊕minus) ⊗ expr
          ⊕              term
term =  Leaf    ® number
      ⊕ middle ® open ⊗ expr ⊗ close
        where middle x y z = y
```