

Addition von Polynomen

```
(define (+terme L1 L2)
  (cond ((leere-termliste? L1) L2)
        ((leere-termliste? L2) L1)
        (else
         (let ((t1 (erster-term L1))
               (t2 (erster-term L2)))
           (cond
            ((> (grad t1) (grad t2))
             (term-anhaengen t1
                              (+terme (rest-terme L1) L2)))
            ((< (grad t1) (grad t2))
             (term-anhaengen t2
                              (+terme L1 (rest-terme L2))))
            (else
             (term-anhaengen
              (konstr-term (grad t1)
                           (add (koeff t1)
                                (koeff t2)))
              (+terme (rest-terme L1)
                       (rest-terme L2))))))))))
```

Multiplikation von Polynomen

```
(define (*terme L1 L2)
  (if (leere-termliste? L1)
      (leere-termliste)
      (+terme (*-term-mit-allen-termen
                (erster-term L1) L2)
        (*terme (rest-terme L1) L2))))
```

```
(define (*-term-mit-allen-termen t1 L)
  (if (leere-termliste? L)
      (leere-termliste)
      (let ((t2 (erster-term L)))
        (term-anhaengen
          (konstr-term (+ (grad t1)
                          (grad t2))
                       (mul (koeff t1)
                              (koeff t2))))
          (*-term-mit-allen-termen
            t1 (rest-terme L))))))
```