

Verarbeitungsphasen eines Interpreters

1. **Lexikalische Analyse:**
kann die eingegebene Zeichenfolge
als Sequenz von Wörtern aufgefasst werden?
 2. **Syntaktische Analyse (Parsing):**
beschreibt die Wortfolge ein Programm?
 3. **Semantische Analyse (Elaboration):**
ist das Programm zulässig?
 4. **Ausführung (Auswertung, Evaluation):**
führe das Programm aus.
- Syntax
- Semantik

Elaboration

1. ein semantisch zulässiges Programm muss **geschlossen** sein:

- ▶ `fun f(x: int) = if x<y then 1 else x*f(x-1)`
offen!
- ▶ `val y=1`
`fun f(x: int) = if x<y then 1 else x*f(x-1)`
geschlossen.

2. Phrasen müssen **wohlgetypt** sein.

- ▶ Typregeln:

$$\frac{e_1 : t_1 \quad o : t_1 * t_2 \rightarrow t \quad e_2 : t_2}{e_1 o e_2 : t}$$

“Der Ausdruck $e_1 o e_2$ ist wohlgetypt und hat Typ t , falls der linke Teilausdruck e_1 den Typ t_1 hat, und der rechte Teilausdruck e_2 den Typ t_2 hat, und der Operator o den Typ $t_1 * t_2 \rightarrow t$ hat.

Semantische Äquivalenz

Zwei Phrasen sind **äquivalent**, wenn sie sich bezüglich statischer und dynamischer Semantik nach außen hin gleich verhalten.

▶ $x*2$ $x+x$

▶ `fun f(x:int)=2+x` `fun f (y:int) = y+2`

Annahme: idealisierter Interpreter!

▶ $(x-y) * (x-y)$ $x*x - 2*x*y + y*y$

