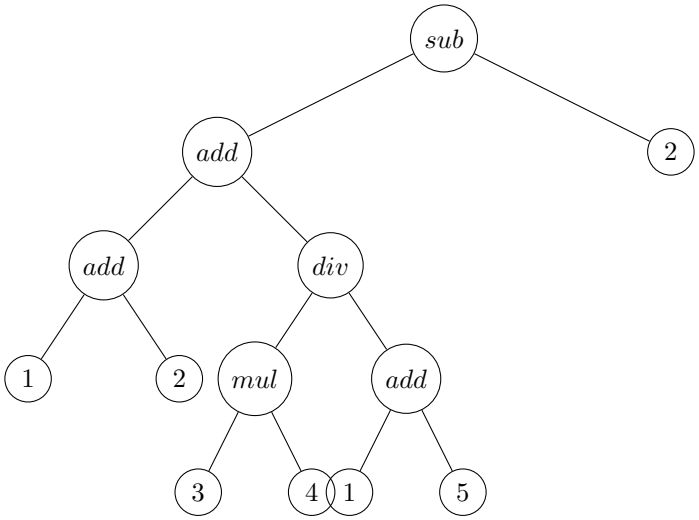
1 Baum

1. Baum besteht aus Knoten (Kreise) und Kanten (Pfeile)
2. Kanten verbinden Knoten mit ihren Kind-Knoten
3. jeder Knoten (außer der Wurzel) hat genau ein Elternteil
4. Knoten ohne Kinder heißen "Blätter“(leaf nodes)
5. Teilbaum
   (a) wähle beliebigen Knoten

Infix Notation:
$1 + 2 + 3 * 4/(1 + 5) - 2$

Präfix Notation:
$sub(add(add(1, 2), div(mul(3, 4), add(1, 5))), 2)$



Präfix Notation aus dem Baum rekonstruieren

1. Wenn die Wurzel ein Blatt ist, dann "Drucke die Zahl“
2. sonst (Operator):
3. Drucke Funktionsnamen
4. Drucke ¨(¨

   $wiederholeab1)frdaslinkeKind$

6. Drucke
7. wiederhole den Algorithmus ab 1) für das rechte Kind
8. Drucke ¨)

Beachte Reihenfolge: Wurzel - Links - Rechts (Pre-Order Traversal) Ergebnis:
$sub(add(add(1, 2), div(mul(3, 4), add(1, 5))), 2)$

Definition: Rekursion Rekursion meint Algorithmus für Teilproblem von vorn

Infix Notation

1. wie bei Präfix
2. sonst
   (a) entfällt
   (b) wie bei Präfix
   (c) wie bei Präfix
   (d) Drucke Operatorsymbol
   (e) wie bei Präfix
   (f) wie bei Präfix
   (g) wie bei Präfix

Beachte Reihenfolge: Links - Wurzel - Rechts (In-Order Traversal)

Ergebnis:
$(((1+2)+((3*4)/(1+5)))+2)$

Berechne den Wert mit Substitutionsmethode

1. Wenn Wurzel ein Blatt hat, gib die Zahl zurück
2. sonst
   (a) entfällt
   (b) entfällt
   (c) wiederhole ab 1) für linken Teilbaum und speichere Ergebnis als $left - result$"
   (d) entfällt
   (e) wiederhole ab 1) für rechten Teilbaum, speichere Ergebnis als $right - result$"
   (f) berechne $fkt_name(left - result, right - result)$ und gib Ergebnis zurück

Beachte Reihenfolge: Links - Rechts - Wurzel (Post-Order Traversal)

$$
\begin{aligned}
& sub(add(add(1,2), div(mul(3,4), add(1,5))), 2) \\
=\, & sub(add(add(1,2), div(12,6)), 2) \\
=\, & sub(add(3,2)2) \\
=\, & sub(5,2) \\
=\, & 3
\end{aligned}
$$

Maschinensprache

- optimiert für die Hardware (viele verschiedene)
- Gegensatz: höhere Programmiersprache ($C++$) ist optimiert für Programmierer
- Compiler oder Interpreter übersetzen Hoch- in Maschinensprache

Vorgang des Übersetzens

1. Eingaben (und Zwischenergebnisse) werden in Speicherzellen abgelegt ⇒ jeder Knoten im Baum bekommt eine Speicherzelle (Maschinensprache: durchnumeriert ; Hochsprache: sprechende Namen)
2. Speicherzellen für die Eingaben *initialisieren* ; Notation: SpZ ← Wert
3. Rechenoperationen in der Reihenfolge des Substitutionsmodells ausführen und in der jeweiligen Speicherzelle speichern ; Notation: SpZ_Ergebnis ← fkt_name SpZ_Arg1 SpZ_Arg2
4. alles in Zahlencode umwandeln
   - Funktionsname ⇒ Opcodes
   - Speicherzellen: nur die Nummer
   - Werte sind schon Zahlen
   - Notation: Opcode Ziel SpZ SpZ_Arg1 SpZ_Arg2 oder Opcode Ziel SpZ Initialwert