

Hinweise zur Programmierung

Unterschiede von `arithmetik2.cpp` ggü. `arithmetik.cpp`:

- ▶ Speichert Eingabezeile mit `getline` auf String `s` ('`\n`' wird automatisch entfernt).
- ▶ Erzeugt zum String `s` den Stringstream `ein`.
- ▶ Liest von Stringstream `ein` statt von Standardeingabe `cin`.
- ▶ Bestimmt mit `ein.tellg` im Fehlerfall die Fehlerposition und markiert diese dann.

Erforderliche Änderungen an `arithmetik2.cpp`:

- ▶ `WertFaktor` ändern, `WertOperand` neu schreiben.
- ▶ Zum Lesen der Funktionsnamen `sin`, `cos`, `exp` und `ln` innerhalb von `WertOperand` die erwartete Zahl von Zeichen lesen, auf einem String speichern und mit Funktionsnamen vergleichen. Falls ungleich oder EOF Fehlerabbruch.

Hinweise zur Programmierung - Fortsetzung

Weitere Anmerkungen:

- ▶ Evtl. eine Funktion schreiben, die die zugelassenen Funktionen auswertet, z.B. mit `WertFunktion("sin")` oder `WertFunktion("ln")` aufgerufen wird, und auch die rechte Klammer im Eingabestrom korrekt liest. Vermeidet Code-Duplikation. (Elegantere Lösung mit `map` und Funktionszeiger → *Programmieren II*)
- ▶ Evtl. Fehlerfunktion mit `string`-Parameter schreiben, um anzuzeigen, in welcher Funktion und an welcher Stelle `fehler` aufgerufen wurde. (Nützlich für Fehlersuche!)
- ▶ Sukzessives Vorgehen: `WertFaktor` zunächst ohne Potenzieren und `WertOperand` ohne Funktionsnamen anlegen, dann Potenzieren und zum Schluss Funktionsauswertungen hinzufügen.