

Praktikumsaufgabe 1 – Taschenrechner

In dieser Aufgabe wird ein einfacher Reverse Polish Notation (RPN) Taschenrechner für das ITS-Board entwickelt. Ein RPN-Rechner arbeitet Stack-basiert. Bei der umgekehrten polnischen Notation werden zunächst die Operanden eingegeben und danach der darauf anzuwendende Operator. Ein binärer Operator wie z.B. + nimmt die beiden obersten Einträge vom Stack, wendet die Operation auf diese an und legt das Ergebnis wieder auf dem Stack ab.

-Modulkonzept und Architektur-

Die Anwendung wurde in folgende Module aufgeteilt, um eine klare Trennung der Verantwortlichkeiten zu gewährleisten.

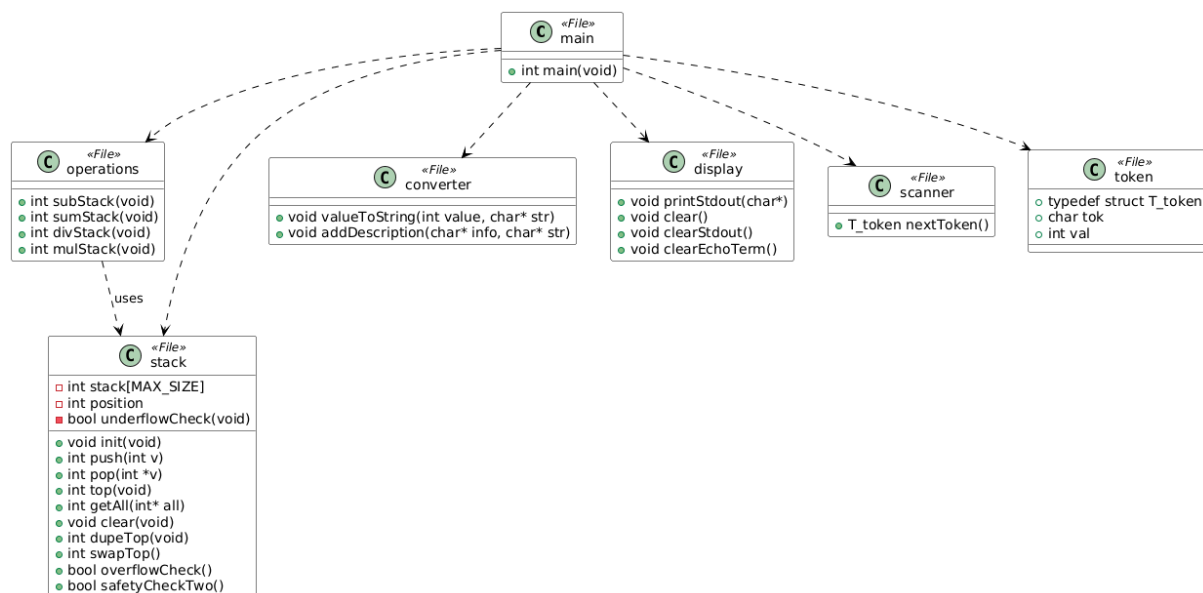


Abb. 1: Modulkonzept des RPN-Taschenrechners mit Abhängigkeiten

operations	Implementierung der arithmetischen Operationen
converter	Konvertierung von Zahlen in Strings für die Anzeige
display	Ausgabe von Ergebnissen und Meldungen auf dem ITS-Board (vorgegeben)
scanner	Einlesen von Benutzereingaben (vorgegeben)
token	Definition der Token-Struktur für die Eingabeverarbeitung
stack	Verwaltung des Operanden-Stacks mit allen Stack-Operationen (push, pop, etc.)

-Fehlerbehandlung-

Für die Fehlerbehandlung wurde ein einheitliches Konzept auf Basis von Return-Codes implementiert. Da C keine Exception-Mechanismen unterstützt, werden Fehler über negative Return-Werte in der Aufrufhierarchie nach oben gereicht.

- 0** Erfolgreiche Ausführung
- 1** Stack-Overflow
- 2** Nicht genügend Operanden auf dem Stack
- 3** Arithmetischer Overflow bei Berechnung
- 4** Stack-Underflow
- 5** Division durch 0

-Sinnvolle Eingabefolgen und Reaktionen-

Eingabe	Beschreibung	Erwartete Ausgabe
3 4 +	Man addiert 3 und 4	7
10 5 -	Man subtrahiert 5 von 10	5
6 7 *	Man multipliziert 6 und 7	42
8 2 /	Man dividiert 8 durch 2	4
5 7 + 3 2 * -	$(5+7) - (3*2)$	6
3 6 8 12 * + / 2 +	$(3 / (6 + (8 + (12*4)))) + 2$	2
2 3 + 5 * 7 -	$((2+3)*5)-7$	18