

Algorithmenbeschreibung mittels Pseudocode

münzausgabe: (IN gezahlt, IN preis, OUT error)

```
differenz := gezahlt - preis; // differenz ist Rückgabebetrag
münzwert := euro_5;
error := FALSE;

REPEAT
    anzahl_münzen (münzwert, differenz, akt_anzahl);
    differenz := differenz - akt_anzahl * Betrag (münzwert); // ermittle Rest
    nächster_münzwert (münzwert, münzwert, fertig); // ermittle nächsten Münzwert
UNTIL fertig;

IF (differenz <> 0)
    THEN error := TRUE;
```

anzahl_münzen: (IN münzwert, IN ausg_betrag, OUT anzahl)

```
berechne_anz := 0;
WHILE differenz >= (berechne_anz + 1) * Betrag (münzwert)
    berechne_anz := berechne_anz + 1; // ermittle Anzahl der Münzen münzwert

münzverwaltung (münzwert, vorhanden); // Münzen münzwert in genügender Zahl vorhanden?
IF vorhanden < berechne_anz
    THEN berechne_anz := vorhanden;

ausgabe (münzwert, berechne_anz); // Münzen ausgeben
anzahl := berechne_anz;
```

nächster_münzwert: (IN alt_münzwert, OUT neu_münzwert, OUT kein_münzwert)

```
kein_münzwert := FALSE;
SWITCH alt_münzwert INTO
    CASE euro_5: neu_münzwert := euro_2;
    CASE euro_2: neu_münzwert := euro_1;
    CASE euro_1: neu_münzwert := cent_50;
    CASE cent_50: neu_münzwert := cent_20;
    CASE cent_20: neu_münzwert := cent_10;
    CASE cent_10: neu_münzwert := cent_5;
    CASE cent_5: neu_münzwert := cent_2;
    CASE cent_2: neu_münzwert := cent_1;
    CASE cent_1: kein_münzwert := TRUE;
```