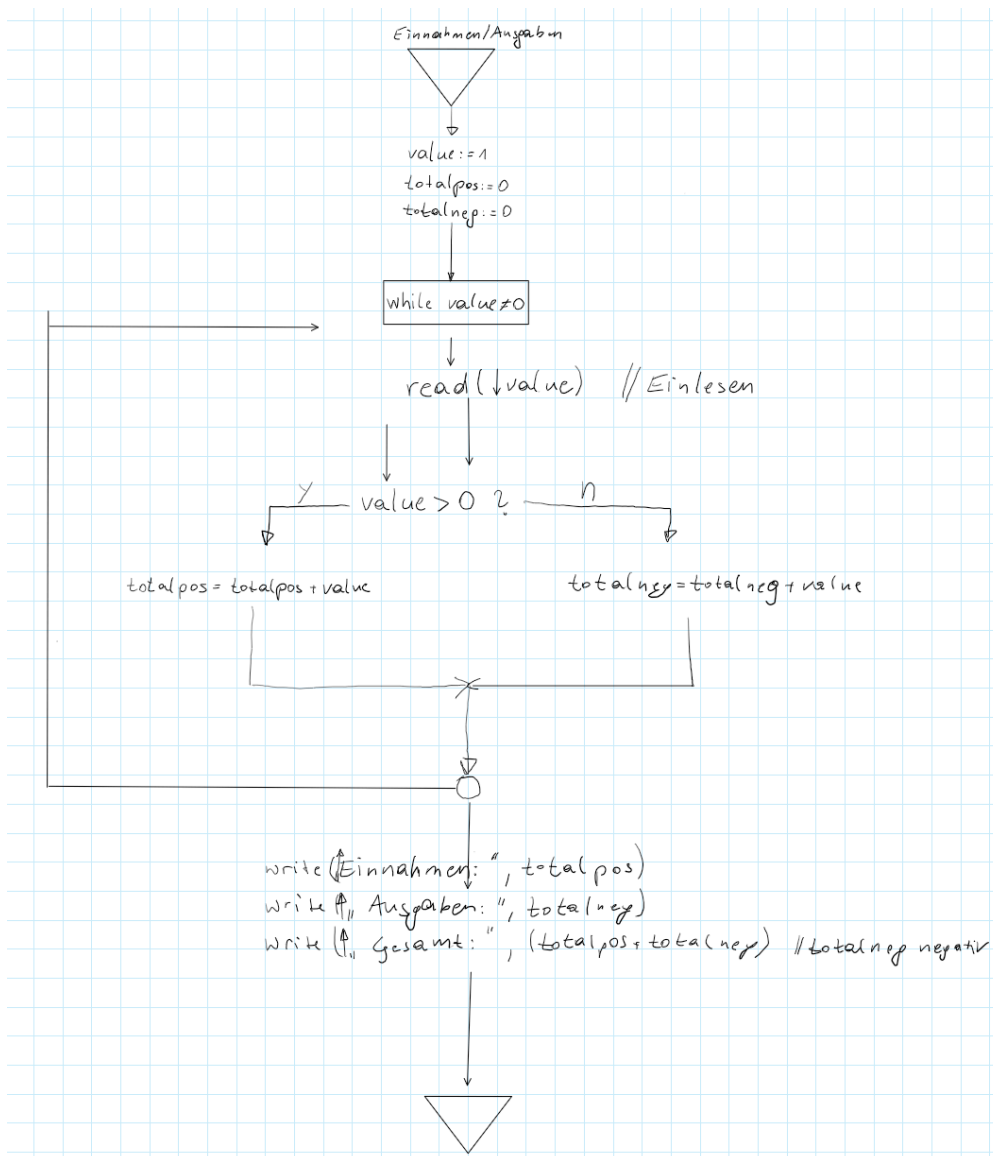


1) Einnahmen / Ausgaben

Lösungsidee

Variablen werden deklariert. Mittels Schleife werden solange Werte eingelesen, bis der eingelesene Wert gleich 0 ist. Alle Werte werden in der Schleife überprüft ob sie größer als 0 sind oder nicht und dann zu den jeweiligen Variablen addiert. Anschließend werden die Werte der Variablen ausgegeben.

a) Ablaufdiagramm



b) Pseudocode

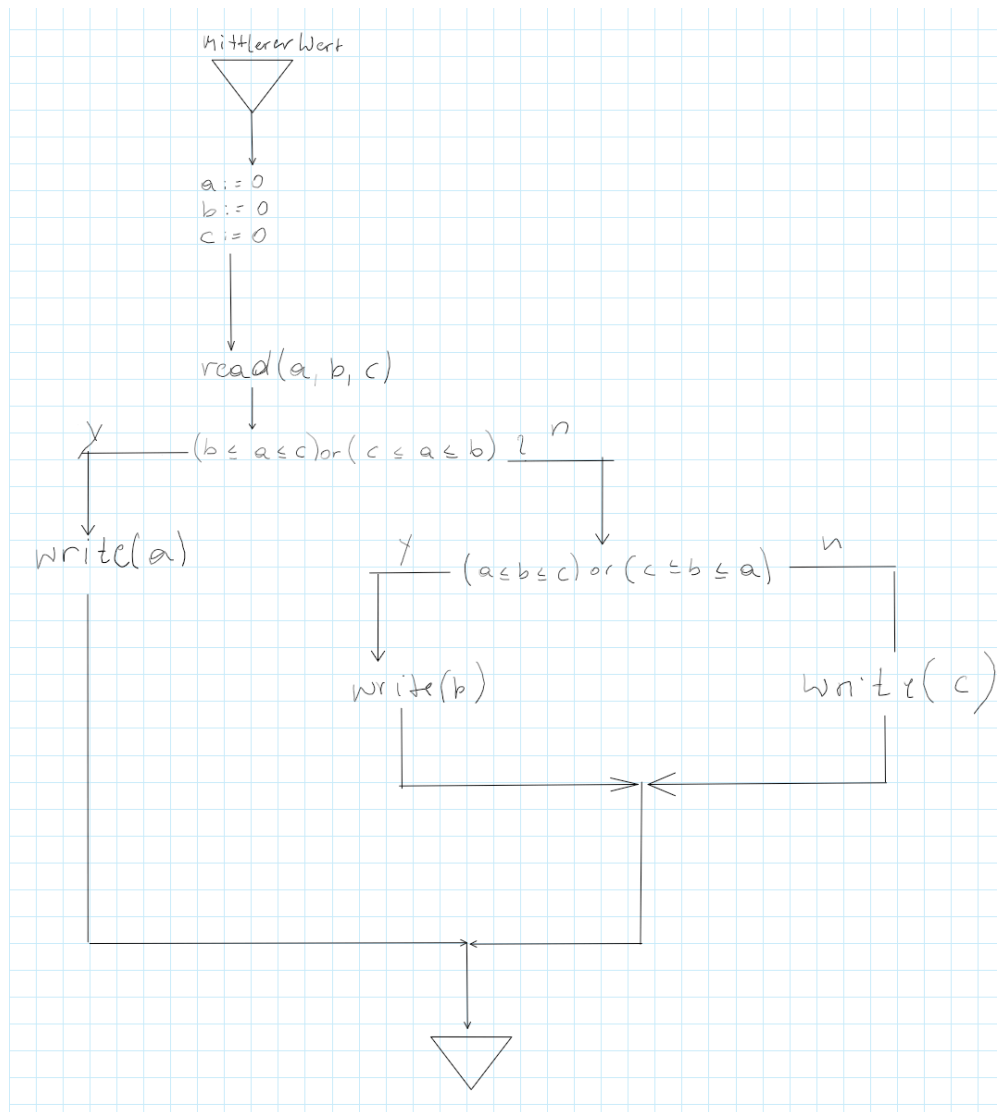
```
1  value := 1
2  totalpos := 0
3  totalneg:= 0
4
5  while value ≠ 0 do
6      read(↑value)
7
8      if value > 0 then
9          totalpos = totalpos + value
10     else
11         totalneg = totalneg + value
12     end if
13 end while
14
15 write(↓"Einnahmen: ", totalpos)
16 write(↓"Ausgaben: ", totalneg)
17 write(↓"Gesamt: ", (totalpos+totalneg)) //totalneg bereits negativ
```

2) Mittelwert

Lösungsidee

3 Variablen werden initialisiert. Anschließend werden drei Werte eingelesen und den Variablen zugeordnet. Durch if-Abfragen und Vergleichen der Werte der Variablen wird der mittlere Wert ermittelt und gleich ausgegeben.

a) Ablaufdiagramm



b) Pseudocode

```
1  a := 0
2  b := 0
3  c := 0
4
5  read(↑a, b, c)
6
7  //1) dry test point
8
9  if (b<=a<=c) or (c<=a<=b) then
10     //2) dry test point
11     write(a)
12 else
13     if (a<=b<=c) or (c<=b<=a) then
14         //3) dry test point
15         write(b)
16     else
17         write(c)
18     end if
19 end if
```

c) Schreibtischtest

Beispiel der Eingabe: 3, 7, 5 // geänderte Reihenfolge für spannenden Schreibtischtest

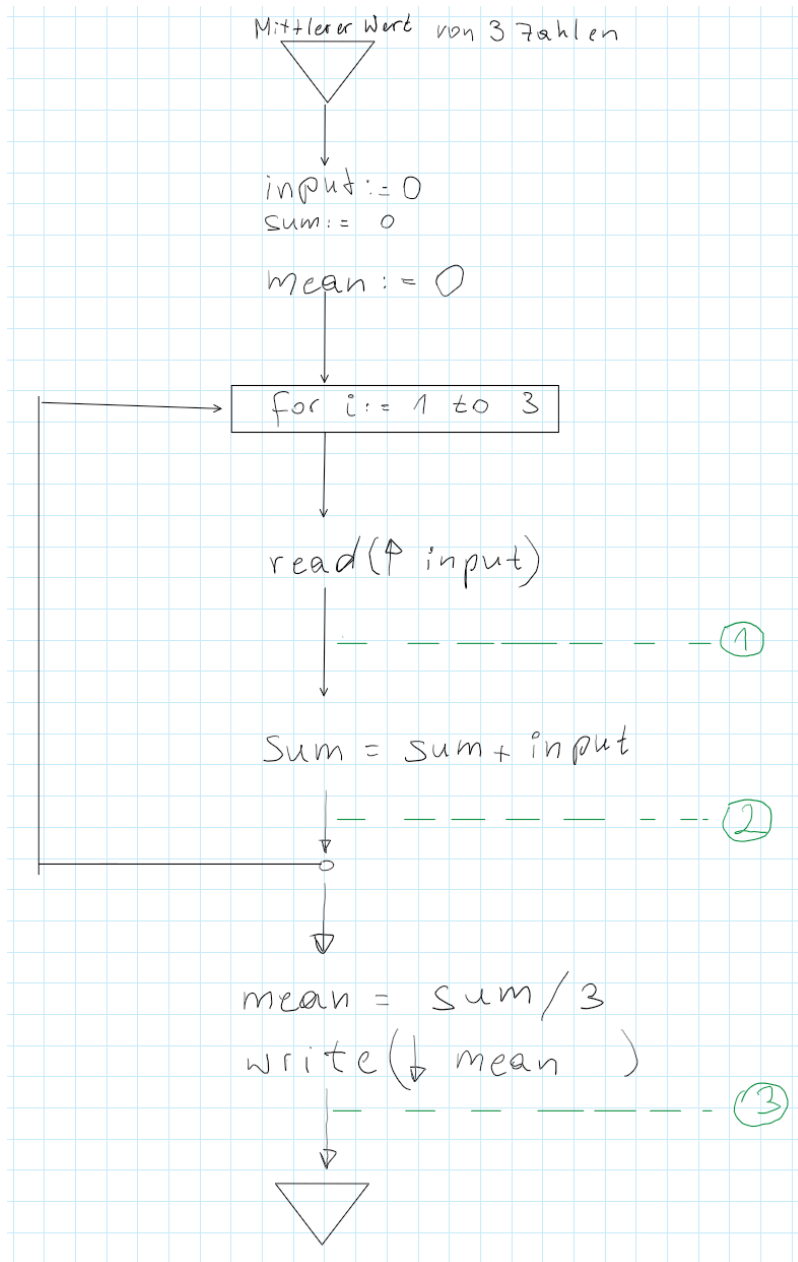
Stelle	a	b	c	(b<=a<=c) or (c<=a<=b)	(a<=b<=c) or (c<=b<=a)	output
1	3	7	5	false	false	-
2	3	7	5	false	false	-
3	3	7	5	false	false	--> 5

2) Mittelwert - Angabe anders verstanden

Lösungsidee:

Variablen werden deklariert. Mittels for-Schleife werden drei Zahlen eingelesen nacheinander aufaddiert. Durch Division mit der Zahl 3 wird das Mittel berechnet. Das Mittel wird ausgegeben.

a) Ablaufdiagramm



b) Pseudocode

```
1  //Mittelwert von 3 Zahlen
2
3  //Initialisierung
4  input := 0
5  sum := 0
6  mean := 0
7
8  for i := 1 to 3 do
9      read(↑input)
10     sum = sum + input
11 end for
12
13 mean = sum/3
14 write(↓mean)
```

c) Schreibtischtest, Input 5, 3, 7

Stelle	input	sum	mean
1	5	0	0
2	5	5	0
1	3	5	0
2	3	8	0
1	7	8	0
2	7	15	0
3	7	15	5**

3) Diskussion: Darstellungsformen

In den Aufgaben 1 und 2 wurden ein Ablaufdiagramm und Pseudocode verwendet. Des Weiteren wurde ein Schreibtischtest zum Testen eines Algorithmus verwendet.

Ablaufdiagramm

Ein Ablaufdiagramm wird als sehr leicht leserlich und verständlich beschrieben. Diese Meinung teile ich grundsätzlich, jedoch bedarf es auch bei dieser Darstellungsform an einer gewissen Gewöhnung. Ich kann mir vorstellen, dass ein Ablaufdiagramm ein wertvolles Tool für die Planung einfacher bis mittelschwerer Algorithmen sein kann, wenn man sich an die Notation gewöhnt.

Für Personen ohne Programmiererfahrung ist es vermutlich einfacher zu verstehen als Pseudocode. Es ist also das Mittel der Wahl, wenn die Logik eines Algorithmus solchen Menschen nähergebracht werden soll.

Pseudocode

Da ich schon etwas Programmiererfahrung sammeln konnte, allerdings noch kein Ablaufdiagramm verwendet habe, war die Darstellung als Pseudocode für mich intuitiver. Pseudocode ist leicht anpassbar und leicht in jedem Texteditor zu erstellen, was möglicherweise einen dynamischeren Workflow ermöglicht.

Pseudocode ist außerdem näher zu Code einer Programmiersprache, wodurch die Umwandlung in der Regel schneller gelingt.