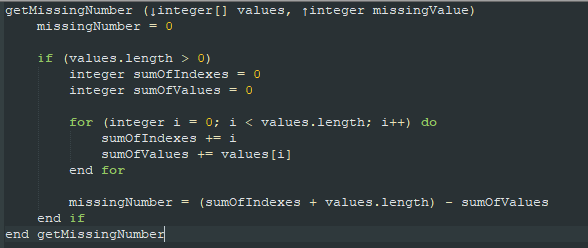
1 „Das fehlende Element“

Lösungsidee:

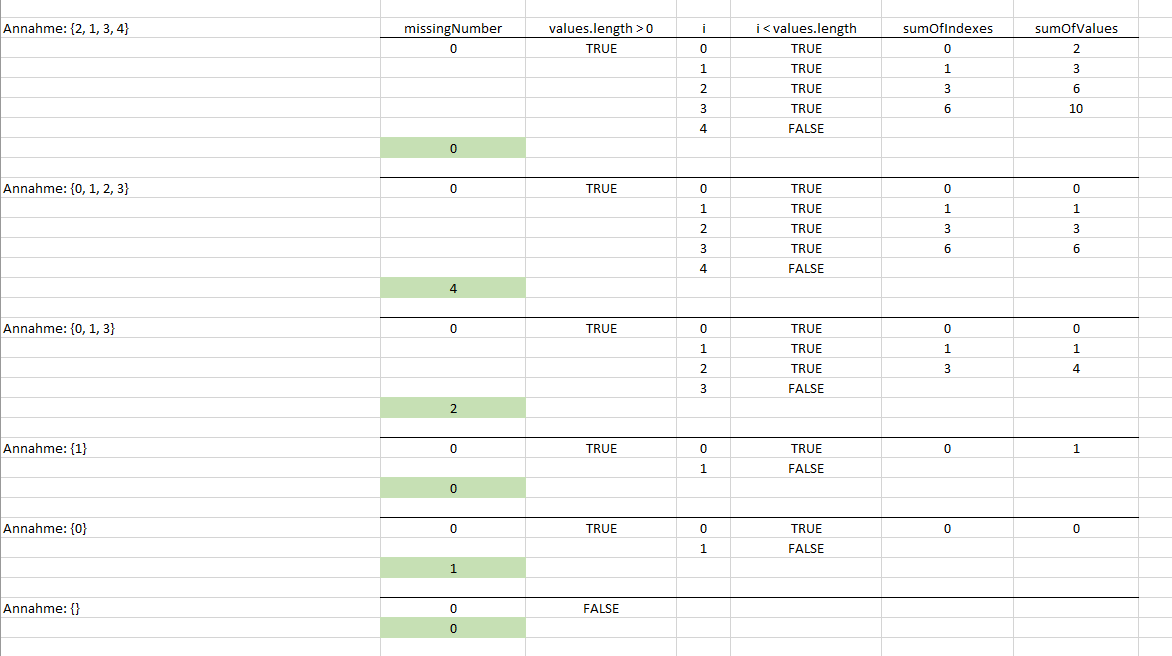
Man addiert alle Werte in dem Array und zeitgleich bildet man eine Summe aus den Indexen vom array (die Zahlen die es sein sollten). Anschließend subtrahiert man von der kompletten summe der Indexes plus der länge des Arrays, die summe des Arrays. Das Ergebnis ergibt die fehlende Zahl.

Testfälle: {2, 1, 3, 4}, {0, 1, 2, 3}, {0, 1 ,3}, {1}, {0}, {}

PseudoCode:



Schreibtischtests:



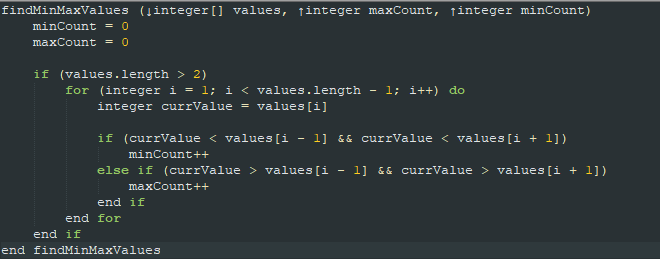
2 Bestimmung von lokalen Minima und Maxima

LösungsIdee:

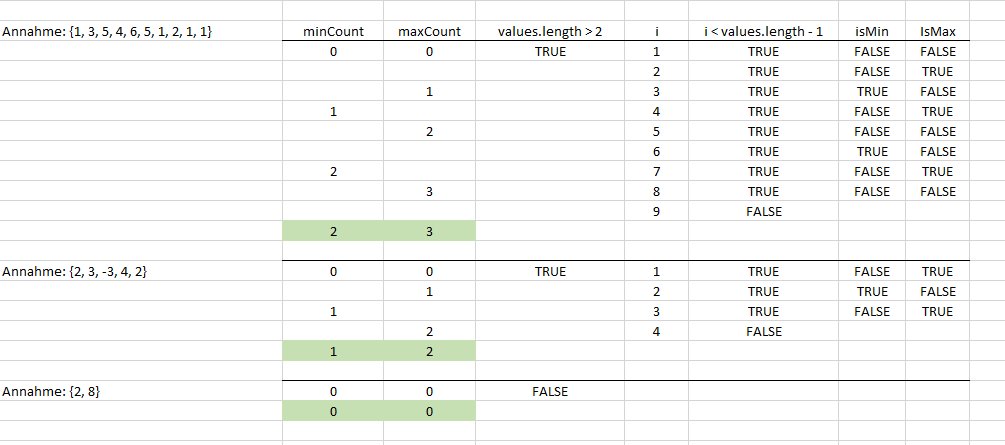
Wenn ein Array n>2 einträge hat läuft ein zähler durch das ganze Array mit Ausnahme der randwerte. An jedem Punkt wird überprüft ob ein lokales Minima oder ein lokales Maxima besteht.

Tesfälle: {1, 3, 5, 4, 6, 5, 1, 2, 1, 1}, {2, 1, -3, 4, 2}, {2, 8}

PseudoCode:



Schreibtischtests:



3 Russische Bauernmultiplikation

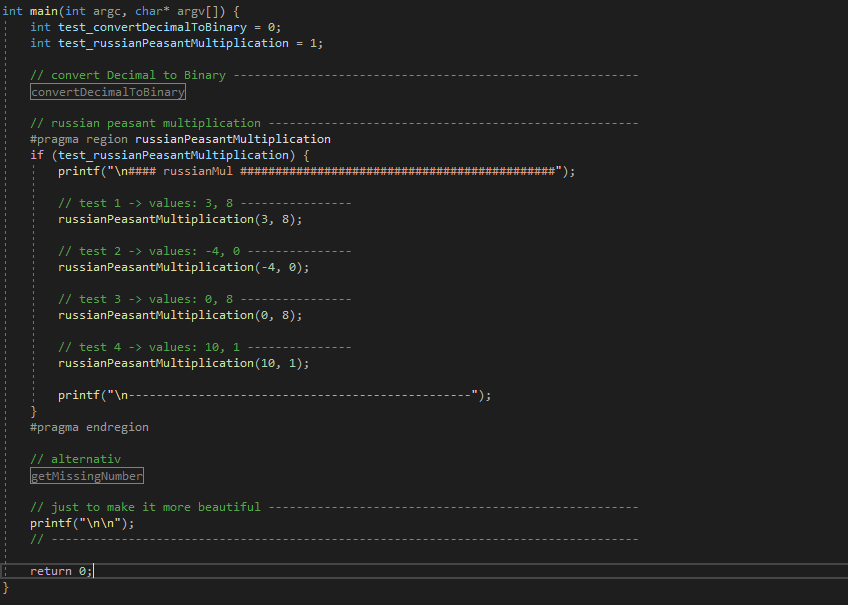
LösungsIdee:

Die Russische Bauernmultiplikation ist ein Multiplikationsverfahren indem der Multiplikator solange halbiert wird bis er 1 erreicht. Der Multiplikant zudem wird immer mit sich selber addiert. Der Multiplikant wird in jedem verlauf, falls der Multiplikator ungerade ist, zu dem endergebnis addiert.

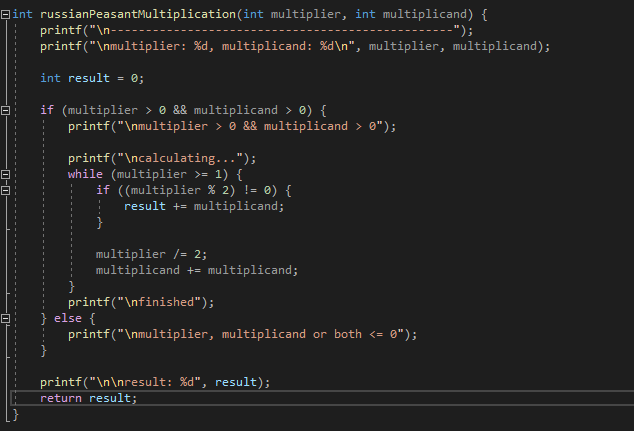
Testfälle: (3, 8) ; (-4, 0) ; (0, 8) ; (10, 1)

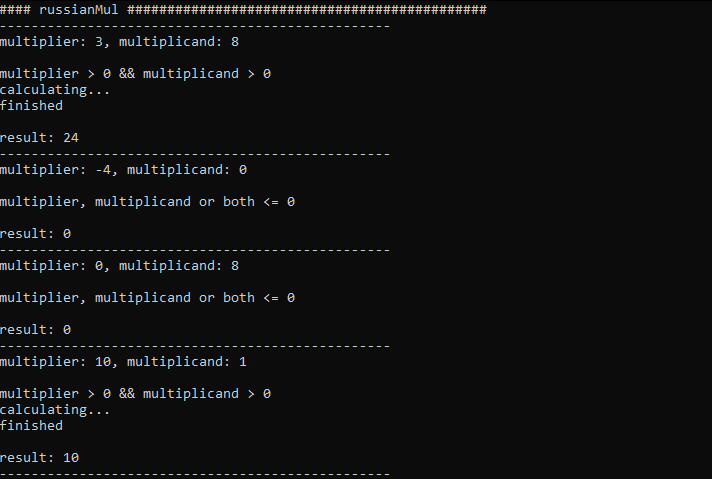
C-Program code:

Int main:



Int russianPeasantMultiplication:



Test Ausgaben: (Start values -> (3, 8), (-4, 9), (0, 8), (10, 1))  
  


4 Umrechnung einer Dezimalzahl in eine Binärzahl

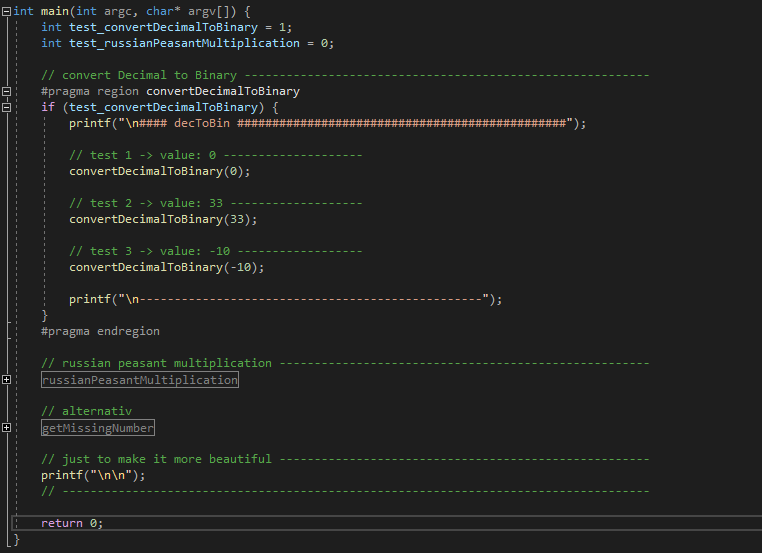
LösungsIdee:

Es wird angenommen, dass die Binärzahl max 64Bit groß ist. Wenn die Dezimalzahl > 0 ist wird das binary Array mit 64 ‚0‘ Einträge erstellt. Zusätzlich wird noch geprüft ob die Dezimalzahl positiv ist.  
Die Dezimalzahl wird anschließend in einer Schleife immer durch 2 geteilt und der Rest in das binary Array gespeichert. Um die gespeicherten Einträge richtig darzustellen wird in einer Schleife das binary Array ‚rückwärts‘ durchloffen und in jeder iteration den aktuellen Array wert augegeben.

Testfälle: 0, 33, -10

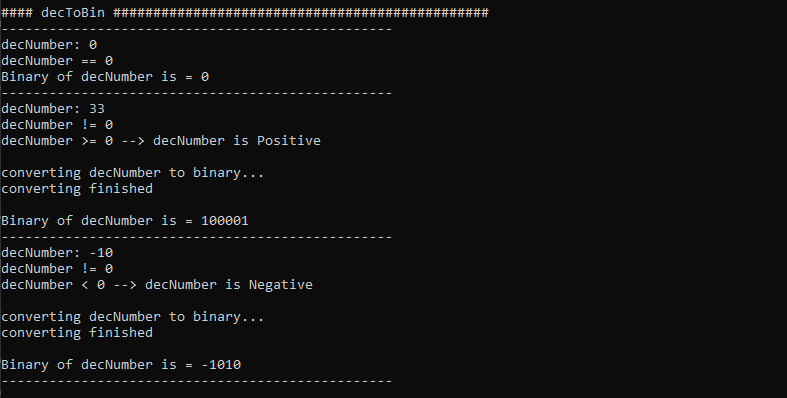
C-Program code:

Int main:



Int convertDecimalToBinary:



Test Ausgaben: (Start values -> 0, 33, -10)