1. Read traverse size (n)
2. Read travers angles {`a[i], i=0..n-1`}
   a) For every angle {`a[i], i=0..n-1`}
   b) Read the angle (`a[i]`)
3. Calculate the actual sum of the angles (`sum`)
   a) Initialize sum to zero (`sum = 0`)
   b) For every angle {`a[i], i=0..n-1`}
   c) Update the sum (`sum = sum + a[i]`)
4. Calculate the correction
   a) The theoritical sum (`tsum = 180 * (n - 2)`)
   b) The error (`error = sum – tsum`)
   c) The correction (`correction = error / n`)
5. Correct the angles
   a) For every angle {`a[i], i=0..n-1`}
   b) Correct the angle (`a[i] = a[i] - correction`)
6. Print the corrected angles
   a) For every angle {`a[i], i=0..n-1`}
   b) Print the angle (`a[i]`)

Write a program that reads traverse angles $a_1, a_2, \ldots, a_n$, performs *Traverse Angle Balancing*, and prints the corrected angles $\hat{a}_1, \hat{a}_2, \ldots, \hat{a}_n$.

$$e = \left(\sum_{i=1}^{n} a_i\right) - 180(n-2)$$

$$c = e/n$$

$$\hat{a}_i = a_i - c \forall i \in [1, n]$$