

Problem D

Combination locks are convenient, because of their design. A combination lock consists of a small wheel to input the combination, and a latch to actually lock things together. The wheel contains consecutive numbers equally spaced around its edge, starting from zero and increasing clockwise. The outer casing of the lock has a small notch. To open the lock, you must rotate the wheel and align the numbers on the wheel with the notch in a specific order.

TopLock, a small company manufacturing combination locks, has recently decided to change its lock setup to give more security than the competitors. Many typical combination locks have a set of three numbers which you must input in order. TopLock locks will have a variable combination length which will be no less than three. In addition, they will be changing the number of numbers on the wheel, which will be between 10 and 100, inclusive.

To open a lock, you must follow the algorithm outlined below:

1. Set the current rotation direction to counterclockwise, and set N to the number of numbers in the combination.
2. Rotate the wheel N full turns in the current rotation direction, where a full turn is a 360 degree spin that leaves the notch pointing at the same number.
3. Continue to rotate the wheel in the current rotation direction until the notch is pointing at the next number in the combination, which successfully inputs that number. Note that by rotating the wheel counterclockwise, the number that the notch points to will increase.
4. If there are no more numbers to input, open the lock, you're finished! Otherwise, reverse the current rotation direction, decrease N by one, and go back to step 2.

It is guaranteed that no two adjacent numbers in combo will be the same, so as to avoid ambiguity in step 3. Also, the starting position of the lock, start, will not be the same as the first number in combo.

Given a `int[] combo`, indicating the combination of the lock, an `int size`, the number of numbers on the wheel, and an `int start`, indicating the number that the notch starts off pointing at, your method should return a double indicating the total number of degrees turned to open the lock.

For example, if `combo = 10,20,30`, `size = 40`, and `start = 6`, you would first rotate the wheel three times counterclockwise and then 4 units to the 10, for a total of $3 * 360 + 4 * (360 / 40) = 1116$ degrees, and then you would rotate twice clockwise and 30 units to the 20, and then once counterclockwise and 10 units to the 30. Your method would in this case return $1116 + 990 + 450 = 2556$.

- The numbers on the combination wheel start at 0 and go to $(size - 1)$, increasing clockwise.
- Double return values must be correct to within an absolute or relative difference of $1e-9$.
- `combo` will have between 3 and 50 elements, inclusive.
- Each element of `combo` will be between 0 and $(size - 1)$, inclusive, and no two consecutive elements will be the same.
- `size` will be between 10 and 100, inclusive.
- `start` will be between 0 and $(size - 1)$, inclusive, and will not be the same as the first element in `combo` (i.e., the one with index 0)

See examples for input/output format.

D.IN

```
1
3
10 20 30
40
6
```

D.OUT

```
2556.0
```