

279 (rotation test) Given two lists, write a program to determine if one list is a rotation of the other. You may use item comparisons, but not list comparisons. Execution time should be linear in the length of the lists.

After trying the question, scroll down to the solution.

§

Can one or both lists be empty? If so, what does it mean for one list to be a rotation of the other? I'll assume the lists are nonempty. Do the lists have to be the same length? Is  $[0; 1; 2; 0; 1; 2]$  a rotation of  $[1; 2; 0]$ ? I'll suppose the two lists have the same length for simplicity. Let the lists be  $A$  and  $B$ , each with length  $n$  where  $n > 0$ . Let  $r$  be a binary variable meaning the two lists are rotations of each other. Then the specification is  $R$ , defined as

$$R = r' = \exists d: 0..n. \forall k: 0..n. A k = B(k \oplus d)$$

where  $\oplus$  is addition modulo  $n$ . This formula suggests looking at displacement  $d=0$ , then displacement  $d=1$ , and so on. Look at this example:

$$A = [0; 0; 0; 0; 0] \quad B = [0; 0; 0; 0; 1]$$

We look at displacement 0 by checking  $A 0 = B 0$ , then  $A 1 = B 1$ , and so on, until  $A 4 = B 4$  is found to be  $\perp$ . So we look at displacement 1 by checking  $A 0 = B 1$ , then  $A 1 = B 2$ , and so on, until  $A 3 = B 4$  is found to be  $\perp$ . And so on. But the execution time is  $n^2$ , not linear in  $n$ . We have to try something else.

I want to rewrite  $R$  so that  $A$  and  $B$  are treated symmetrically.

$$R = r' = \exists i, j: 0..n. \forall k: 0..n. A(i \oplus k) = B(j \oplus k)$$

or, equivalently,

$$R = r' = \exists i, j: 0..n. \forall k: 0..n. (A;;A)(i+k) = (B;;B)(j+k)$$

That is a bit bizarre, but it is equivalent to the previous definition of  $R$ . Now define  $a$  to be the maximum list (in list order) of all rotations of  $A$ . Define  $b$  similarly.

$$a = \uparrow k: 0..n. A[(k;..n); (0;..k)]$$

$$b = \uparrow k: 0..n. B[(k;..n); (0;..k)]$$

or, equivalently,

$$a = \uparrow k: 0..n. (A;;A)[k;..k+n]$$

$$b = \uparrow k: 0..n. (B;;B)[k;..k+n]$$

Now define  $Q$  to say that all rotations of  $A$  starting before index  $i$  are less than  $b$ , and symmetrically that all rotations of  $B$  starting before index  $j$  are less than  $a$ .

$$Q = (\forall k: 0..i. A[(k;..n); (0;..k)] < b) \wedge (\forall k: 0..j. B[(k;..n); (0;..k)] < a)$$

or, equivalently,

$$Q = (\forall k: 0..i. (A;;A)[k;..k+n] < b) \wedge (\forall k: 0..j. (B;;B)[k;..k+n] < a)$$

And finally, let  $P$  say that a segment of  $A$  starting at  $i$  of length  $h$  (wrapping around if necessary) equals a segment of  $B$  starting at  $j$  of length  $h$  (wrapping around if necessary).

$$P = (A;;A)[i;..i+h] = (B;;B)[j;..j+h]$$

Now the problem is solved as follows.

$$R \iff i:=0. j:=0. i < n \wedge j < n \wedge Q \implies R$$

$$i < n \wedge j < n \wedge Q \implies R \iff h:=0. i < n \wedge j < n \wedge Q \wedge h < n \wedge P \implies R$$

$$i < n \wedge j < n \wedge Q \wedge h < n \wedge P \implies R \iff$$

$$\text{if } A(i \oplus h) < B(j \oplus h) \text{ then } i:=i+h+1. j < n \wedge Q \implies R$$

$$\text{else if } A(i \oplus h) > B(j \oplus h) \text{ then } j:=j+h+1. i < n \wedge Q \implies R$$

$$\text{else } h:=h+1. i < n \wedge j < n \wedge Q \wedge h \leq n \wedge P \implies R \text{ fi fi}$$

$$j < n \wedge Q \implies R \iff \text{if } i \geq n \text{ then } s:=\perp \text{ else } i < n \wedge j < n \wedge Q \implies R \text{ fi}$$

$$i < n \wedge Q \implies R \iff \text{if } j \geq n \text{ then } s:=\perp \text{ else } i < n \wedge j < n \wedge Q \implies R \text{ fi}$$

$$i < n \wedge j < n \wedge Q \wedge h \leq n \wedge P \implies R \iff$$

$$\text{if } h=n \text{ then } s:=\top \text{ else } i < n \wedge j < n \wedge Q \wedge h < n \wedge P \implies R \text{ fi}$$

The execution time bound  $3 \times n$  is easily proven, but I think maybe  $2 \times n$  is possible.