CS580
Homework 05:
Transpose.
Yie Sheng Chen 1303

Problem definition

Consider a square matrix B of size N. Let A represent the initial configuration of the matrix B.

(1) A UNITY program is developed that transposes the rows and columns of matrix B and preserves the following invariant:

```
inv. p≤q ^
     < ∀ i,j : (1≤i<p v 1≤j<p v q<i≤N v q<j≤N) ^ 1≤i≤N ^ 1≤j≤N ::
         B(i,j) = A(j,i) >
```

(2) Write a formal specification of the correctness of the program you designed. Such a specification often assumes the following general form:

a. init -> Post b. stable Post

(3) Explain in narrative form (no formal proof) the steps involved in proving these two properties.

(1) UNITY program

```
Program Transpose
declare
    B    : array[1..N, 1..N] of integer
initially
    B = A [] p = 2 [] q = N-1
assign
    p := p+1 if p<q || q := q-1 if p+1<q
    ||
    <|| i,j : (p-1≤i<p v p-1≤j<p v q<i≤q+1 v q<j≤q+1 ) ^ 1≤i≤N ^ 1≤j≤N ^ p!=q::
           B(i,j),B(j,i) := B(j,i),B(i,j)>
end
```

(2) formal spec.

```
init leads-to post
stable post

init ≡ p = 2 ^ q = N-1 ^ B = A
post ≡ < ∀ i,j : 1≤i≤N ^ 1≤j≤N :: B(i,j) = A(j,i) > ^ p = q
```

(3) narrative explanation of steps in proving properties stated in (2)

1. stable post

```
Assume post "< ∀ i,j : 1≤i≤N ^ 1≤j≤N :: B(i,j) = A(j,i)> ^ p = q"
is true

Looking at the only statement in the UNITY program Transpose,
if p = q, both "p<q" and "p+1<q" does not hold,
so p and q's value will not change.

And if p =q, "B(i,j),B(j,i) := B(j,i),B(i,j)" will not take place.
Therefore, B(i,j) does not change
and "B(i,j) = A(j,i)" will continue to hold.
```

2. init leads-to post

From "initially" section of the UNITY program,
"init ≡ p = 2 ^ q = N-1 ^ B = A" holds.

select the well-founded metric to be:
X = the number of elements (i,j) in array
    that "B(i,j) = A(j,i)" does not hold

This metric is well-founded because its minimum value is 0

And X decreases which can be proved from
{X=k} ensures {X<k}
since by looking at the only statement in the UNITY program Transpose,
before p=q holds, at each step,
this statement transposes the array elements
in the darker shaded region and thus decreases X.