Assignment 3

Task 3: Loops & Numerical Computation

The goal is to prove the method mult(x, y) returns $x * y$ for all $x, y \geq 0$

Loop invariant:

$x * y == k * n + res$

before the loop starts, we have

$k = x$

$n = y$

$res = 0$

which means:

$k * n + res = x * y + 0 = x * y$

the loop invariant ($x * y == k * n + res$) holds before the first iteration

there are two cases

Case 1: k % 2 == 1 (Odd)

Let

k_old = k, n_old = n, res_old = res

After going that if:

res = res + n, k = k / 2, n = 2 * n

k_new * n_new + res_new

= (k_old / 2) * (2 * n_old) + (res_old + n_old)

= k_old * n_old + res_old + n_old

= x * y - n_old + n_old

= x * y

Case 2: k % 2 == 0 (Even)

res remains unchanged

k = k / 2, n = 2 * n

k_new * n_new + res_new

= (k_old / 2) * (2 * n_old) + res_old

= k_old * n_old + res_old

= x * y

The loop will terminate when k ==0

x * y == k * n + res

=> x * y == 0 * n + res

=> x * y == res

This matches the postcondition

// post-condition: returns x * y

Task 4: Looping an Array

Precondition:

1. A is not Null (A != null)
2. X is any integer
3. A.length >= 0

Postcondition:

1. if x appears in the array A, return the last index i that A[i] == x.
2. if x doesn’t appear in the array, then just return -1

Loop invariant

We iterate from the back of the array (i = n- 1) to the beginning (i=0) so

For all indices j that i < j < A.length, A[j] != x

By which I mean is that: Every element to the right of I has been checked but did not contain x

So, this ensures that when A[i] == x, it is the last occurrence.

So, we begin with

int n = A.length, i = n - 1;

at this point the range i < j < A.length is empty (no j satistfies this condition) so the invariant is fine

There are 2 cases

Case 1: A[i] == x

The function returns i

By the invariant, we know all j > i do not contain x, so i is the last occurrence.

This satisfies the postcondition

Case 2: A[i] != x

Decrement will happen: i = i -1

We then can add i + 1 into the checked indices

Since A[i + 1] != x, the invariant is maintained for the new i

So invariant still maintained

Termination:

Loop will end when i < 0 which means all indices 0 ≤ j < A.length have been checked, and none of them equal x

The function then returns -1

Poscondition is satisfied when x is not found

Chaiyanun Sakulsaowapakkul 6681299