ICCS208: Assignment 3

Theeradon Sarawek
theeradon.sar@student.mahidol.edu
14 October 2024

1: Task 3: Loops & Numerical Computation

Consider the following code, now with filled-in Hoare logic justifications:

```
// precondition: x >= 0 && y >= 0;
int mult(int x, int y) {
    int k = x, n = y, res = 0;
    while (k != 0) { // @loop_invariant x * y == k * n + res;
        // { k = k0, n = n0, res = 0 for some k0, n0 }
        if (k%2 == 1) {
            // { k%2 == 1 } and {  x * y == k0 * n0 + res }
            res = res + n;
            // {  x * y == k0 * n0 + res + n0 }
        }
        k /= 2;
        // {  x * y == (k0/2) * n0 + res }
        n *= 2;
        // {  x * y == k0 * (n0*2) + res }
    }
    return res;
    // { res == x * y }
}
// post-condition: returns x * y
```

2: Task 4: Looping an Array

Consider the following code, now with filled-in Hoare logic justifications as well as a precondition + postcondition:

```
// precondition: A.length >= 0, x is an integer
int find(int[] A, int x) {
    int n = A.length, i = n - 1; // { n = A.length >= 0 => i = n -
       1 >= 0 }
    while (i >= 0) { // @loop_invariant {i >= -1} & {n = A.length}
        if (A[i] == x)
            // { A[i] == x }
            return i;
            // {0 <= i <= n - 1} postcondition #1
        i -= 1;
        // {i = i - 1 => i >= -1}
    }
    // { i == -1 } postcondition #2
```

```
    return -1;
}
// postcondition: return index i where {0 <= i <= n - 1} of first
   occurence of x counting backwards, otherwise return -1 for no
   occurences.
```