

# Software Specifications

## Assignment 4

Cain Susko

Queen's University  
School of Computing

March 23, 2022

1. The Precondition for the given statements should be:

- (a)  $x == 1$
- (b)  $x == 2$
- (c)  $0 < 2y + 2z$
- (d)  $2(y*z + 3) > y + 2$
- (e)  $\text{Exists}(y = 0; y < 10) \ 2y + 1 == 50$
- (f)  $\text{Exists}(y = 0; y < 15) \ 2(x+y) == y + t$
- (g)  $\text{ForAll}(z = 1; z < 100) \ 3y + 1 > z + 2$
- (h)  $\text{ForAll}(y = 1; y < x) \ 2x + 2y < 100$
- (i)  $\text{Exists}(y+z = 0; y+z < 10) \ z*z + 2(y + z) == 15$
- (j)  $\text{Exists}(y = 0; y < 100) (2y + z == 15 \ || \ 2z*y + y < 100)$

2. These following statements have been verified:

- (a) 

```
ASSERT( x >= 5 || ( x <= 0 && y == 2 ) )
z = x - y;
ASSERT(y == 2)
y = y + z;
ASSERT(y - z == 2 || y > 3)
x = y - z;
ASSERT( x == 2 || y > 3 )
```
- (b) 

```
ASSERT( z >= 0 )
if ( x < 2 ) { y = z+3; } //end-if
ASSERT(y > x + 1 || x + z > 1)
z = x + z;
ASSERT(y > x + 1 || z > 1)
```

3. (a) The Loop Invariant is:  $i > 0$

(b) The Complete Proof Tableau is:

```

ASSERT(k >= 0)
i = k;
ASSERT(i > 0)
sum = k;
while( i > 0 )
{
    ASSERT(sum+i-1 == j && i > 0)
    i = i - 1;
    ASSERT(sum+i == j && i > 0)
    sum =sum+i;
}
ASSERT(sum == SUM{j=0->k}(j))

```

```

4. const int n; /* the program will
                compute the sum of the squares
                of the n smallest positive odd integers*/
int sum;      /* the sum is stored in this variable */

include <math.h>
int count

ASSERT( n >= 1 )

for (int i=0; count <= n; i++){
    ASSERT(count <= n)
    if (i % 2 != 0){
        ASSERT(sum % 2 == 0 && pow(i, 2) ==
                (2*count)*(2*count))
        count++;
        ASSERT(sum % 2 == 0 && pow(i, 2) ==
                (2*count+1)*(2*count+1))
        sum += pow(i, 2);
    }
}
ASSERT( sum == SUM{i=0->n-1} (2*i+1)*(2*i+1) )

```