Homework 19 Solutions
CS 336

The important issue is the logic you used to arrive at your answer.

1. Prove that the program segment

```
fac := 1
i := 2
while i ≤ n do
    fac := fac*i
    i := i+1
```

is partially correct with respect to precondition "$n \geq 1$ " and postcondition " $fac = n!$ ".
Be explicit about your loop invariant: I = " $fac = (i-1)! \wedge i \leq n+1$"

	$n \geq 1$
fac := 1	$n \geq 1 \wedge fac = 1$
i := 2	$n \geq 1 \wedge fac = 1 \wedge i = 2$
	$fac = (i-1)! \wedge i \leq n+1$
while i ≤n **do**	$fac = (i-1)! \wedge i \leq n$
fac := fac*i	$fac = i! \wedge i \leq n$
i := i+1	$fac = (i-1)! \wedge i \leq n+1$
	$fac = (i-1)! \wedge i \leq n+1 \wedge i > n$
	$i = n+1 \wedge fac = n!$
	$fac = n!$

And this loop terminates because

	true
fac := fac*i	*true*
i := i+1	$n\text{-}i < n\text{-}i'$

Thus since the integer values of n-i must strictly descend after each execution of the loop, eventually we must have n-i<0 and the loop terminates.

2. Prove that the program segment

```
i := 1
while key_i ≠ test do
    i := i+1
```

is partially correct with respect to precondition " $(n \geq 1) \wedge (\exists i \ni 1 \leq i \leq n \wedge key_i = test)$ " and postcondition " $(key_i = test) \wedge (1 \leq j < i \Rightarrow key_j \neq test)$ ".

Be explicit about your loop invariant: I = " $1 \leq j < i \Rightarrow key_j \neq test$ "

_________________________ $(n \geq 1) \wedge (\exists i \ni 1 \leq i \leq n \wedge key_i = test)$

i := 1 _________________________ $1 \leq j < i \Rightarrow key_j \neq test$

while $key_i \neq test$ **do** ____________ $1 \leq j < i+1 \Rightarrow key_j \neq test$

i := i+1 ____________________ $1 \leq j < i \Rightarrow key_j \neq test$

_________________________ $(key_i = test) \wedge (1 \leq j < i \Rightarrow key_j \neq test)$

And this loop terminates because

_________________________ $(n \geq 1) \wedge (\exists j \ni 1 \leq j \leq n \wedge key_j = test)$

i := 1____________________ $(i = 1) \wedge (n \geq 1) \wedge (\exists j \ni 1 \leq j \leq n \wedge key_j = test)$

_________________________ $\exists j \ni i \leq j \leq n \wedge key_j = test$

while $key_i \neq test$ **do**

_____________________ $(\exists j \ni i \leq j \leq n \wedge key_j = test) \wedge (key_i \neq test)$

_____________________ $\exists j \ni i+1 \leq j \leq n \wedge key_j = test$

i := i+1_______________ $(\exists j \ni i \leq j \leq n \wedge key_j = test) \wedge (n - i' < n - i)$

The quantity n-i is strictly decreasing. Should it ever become negative then the assertion $\exists j \ni i \leq j \leq n \wedge key_j = test$ would be falsified. Yet this assertion is true at the top of every pass through the loop. Therefore, there must be termination.

3. Prove the following code is partially correct with respect to precondition "$n \geq 1$" and postcondition "$(k/2 < n) \wedge (k \geq n) \wedge (\exists j \geq 0 \ni k = 2^j)$" (assume k and n are integer variables.):

```
k := 1
while k < n  do
        k := 2*k
endwhile
```

Be explicit about your loop invariant: I = $(k/2 < n) \wedge (\exists j \geq 0 \ni k = 2^j)$

______________________ $n \geq 1$

k := 1

______________________ $(n \geq 1) \wedge (k = 1)$

______________________ $(k/2 < n) \wedge (\exists j \geq 0 \ni k = 2^j)$

while k < n **do**

_______________ $(k/2 < n) \wedge (\exists j \geq 0 \ni k = 2^j) \wedge (k < n)$

_______________ $(k < n) \wedge (\exists j \geq 0 \ni k = 2^j)$

k := 2*k

_______________ $(k' < n) \wedge (\exists j \geq 0 \ni k' = 2^j) \wedge (k = 2k')$

_______________ $(k/2 < n) \wedge (\exists j \geq 0 \ni k = 2^j)$

endwhile

______________________ $(k/2 < n) \wedge (k \geq n) \wedge (\exists j \geq 0 \ni k = 2^j)$

And this loop terminates because

_________________________ $k \geq 1$

k := 2*k_____________________ $k' \geq 1 \wedge k = 2k$

_________________________ $k \geq 1 \wedge n - k < n - k'$

Thus since the integer values of *n-k* must strictly descend after each execution of the loop, eventually we must have $n - k \leq 0$ and the loop terminates.