

## Homework Assignment #4

**Due Time/Date**

2:20PM Wednesday, November 4, 2020. Late submission will be penalized by 20% for each working day overdue.

**Note**

Please write or type your answers on A4 (or similar size) paper. Put your completed homework on the instructor's desk before the class starts. For late submissions, please drop them in Yih-Kuen Tsay's mail box on the first floor of Management Building 2. You may discuss the problems with others, but copying answers is strictly forbidden.

**Problems**

We assume the binding powers of the logical connectives and the entailment symbol decrease in this order:  $\neg$ ,  $\{\forall, \exists\}$ ,  $\{\wedge, \vee\}$ ,  $\rightarrow$ ,  $\leftrightarrow$ ,  $\vdash$ .

1. Prove that the following annotated program segments are correct:

- (a) (10 points)

```
{true}
if  $x < y$  then  $x, y := y, x$  fi
 $\{x \geq y\}$ 
```

- (b) (10 points)

```
 $\{g = 0 \wedge p = n \wedge n \geq 1\}$ 
while  $p \geq 2$  do
   $g, p := g + 1, p - 1$ 
od
 $\{g = n - 1\}$ 
```

- (c) (20 points) For this program, prove its total correctness.

```
 $\{y > 0 \wedge (x \equiv m \pmod{y})\}$ 
while  $x \geq y$  do
   $x := x - y$ 
od
 $\{(x \equiv m \pmod{y}) \wedge x < y\}$ 
```

2. A majority of an array of  $n$  elements is an element that has more than  $\frac{n}{2}$  occurrences in the array. Below is a program that finds the majority of an array  $X$  of  $n$  elements or determines its non-existence. (Hint: if  $A[i] \neq A[j]$ , then the majority of  $A$  remains a majority in a new array  $B$  obtained from  $A$  by removing  $A[i]$  and  $A[j]$ . Check out Udi Manber's algorithms book if you cannot understand the program.)

```

C,M := X[1],1;
i := 2;
while i<=n do
  if M=0 then C,M := X[i],1
    else if C=X[i] then M := M+1
      else M := M-1
    fi
  fi;
  i := i+1
od;
if M=0 then Majority := -1
  else Count := 0;
  i := 1;
  while i<=n do
    if X[i]=C then Count := Count+1 fi;
    i := i+1
  od;
  if Count>n/2 then Majority := C
    else Majority := -1
  fi
fi

```

- (a) (30 points) Annotate the program into a *standard* proof outline, showing clearly the partial correctness of the program; a standard proof outline is essentially an annotated program where every statement is surrounded by a pair of pre- and post-conditions.
- (b) (30 points) Prove the validity of the annotation for the first while loop.