Homework 1 Solutions

1. a) Note that since A, B, and C can be arbitrary C expressions, we must ensure that we execute them only if they would have been executed in C. Imagine that B is equal to "x/2 == 1" and you should see why we must use a nested if.

```
if  A -> s1;
         if ~B -> s2
         []  B -> if  C -> s3
                  [] ~C -> s4
                  fi
         fi
[] ~A -> SKIP
fi
```

b)

```
i:=0;
do i<n -> j:=i;
          do j<n -> k:=n-i;
                    j:=j+1;
          od
          i:=i+1
od
```

c) Here I am introducing a new variable "_cont" which assumes that this variable has not already been used.

```
_cont:=1
do _cont!=0 -> s1;
               if  C -> _cont:=0
               [] ~C -> s2;
                        if  A -> if  B ->_cont:=1
                                 [] ~B ->_cont:=0
                                 fi
                        [] ~A -> _cont:=0
                        fi
               fi
od
```

2. a) Given this implementation of assignment, both the l-value and the r-value of A[i] will be computed before i is modified.

b) When A[i] $\neq$ A[i+1]

3. a) 68 34

b) 68 61

4. a) 20, 61 20

b) 58, 58 58

c) 20, 61 61

d) 58, 58 58