Burt Rosenberg
Final Exam

1. How do these two code blocks differ:

```c
/* First code block */
i = 0 ;
while (i*i<N) {
i = i+1 ;
}

/* Second code block */
i = 0 ;
do {
i = i+1 ;
} while (i*i<N) ;
```

**Hint:** They differ only for certain values of $N$.

2. Logical equivalence of formulas can be useful to simplify your code. Show that these two code blocks function equivalently:

```c
/* First code block */
if( !a ) {
    if ( !b ) {
        printf("The condition on a and b is satisfied.\n") ;
    }
}

/* Second code block */
if ( a || b ) {
}
else {
    printf("The condition on a and b is satisfied.\n") ;
}
```

**Hint:** Use logical deduction to show that, as a function of the integer variables $a$ and $b$, the `printf` is run in the first code block only if for the same values of $a$ and $b$, it would run in the second code block.
3. Although logical equivalent, the following two code blocks do not function equivalently. State why.

/* First code block */
if ( a ) {
    if ( b ) {
        printf("The condition on a and b is satisfied.\n") ;
    }
}

/* Second code block */
if ( b && a ) {
    printf("The condition on a and b is satisfied.\n") ;
}

4. What does this program do?

```c
int mystery( int n, int m) {
    int j, k, l ;
    if ( n<=0 ) { return(0) ; }
    k = 0 ;
    j = 0 ;
    if ( n%2 ) {
        j = m ;
        k = 1 ; }
    l = mystery( (n-k)/2,m ) ;
    return( l+l+j ) ;
}
```

5. For every integer $N$, does this program eventually print out “Done” then stop?

```c
#include<stdio.h>
#define N 10
int main(int argc, char * argv[] ) {
    int n ;
    n = N ;
    while ( n!= 1 ) {
        if ( n%2 ) { n = 3*n-1 ; }
        else n = n/2 ;
    }
    printf("Done\n") ;
}
```