COMP 3500: Homework 1

- Jacob Howard
- JAH0147
- 903937044

Questions:

1. [60 points] Consider the following program:

```
P1: {
  shared int x;
  x = 10;
  while (1) {
     x = x - 1;
     x = x + 1;
     if (x != 10)
       printf("x is %d",x)
  }
}
```

```
P2: {
  shared int x;
  x = 10;
  while ( 1 ) {
    x = x - 1;
    x = x + 1;
    if (x!=10)
      printf("x is %d",x)
  }
}
```

Note that the scheduler in a uniprocessor system would implement pseudo parallel execution of these two concurrent processes by interleaving their instructions, without restriction on the order of the interleaving.

1.1. [25 points] Show a sequence (i.e., trace the sequence of interleavings of statements) such that the statement "`x is 10`" is printed.

1.	x = x-1; //x = 9
2.	X = x +1; //x = 10
3.	X = x-1; //x = 9
4.	If (x != 10) //9
5.	X = x + 1; //x = 10
6.	Printf("x is %d", x); *//10*
6.	"x is 10" is printed

1.2. [35 points] Show a sequence such that the statement "`x is 8`" is printed.

```
LD R0, x 10 10 –
DECR R0 10 9 –
STO R0, x 9 9 –
LD R0, x 9 9 9
DECR R0, x 9 9 8
STO R0, x 8 9 8
LD R0, x 8 8 8
INCR R0 8 9 –
LD R0, x 898
INCR R0 8 9 9
STO R0, x 9 9 9

If (x != 10) printf("x is %d", x);
"x is 9" is printed

STO R0, x 9 9 9

If (x != 10) printf("x is %d", x);
"x is 9" is printed

LD R0, x 9 9 9
DECR R0 9 8 –
STO R0, x 8 8 –
LD R0, x 8 8 8
DECR R0 8 8 7
STO R0, x 7 8 7
LD R0, x 7 7 7
INCR R0 8 8 7
STO R0, x 8 8 7

If (x != 10) printf("x is %d", x);
"x is 8" is printed
```

You should remember that the increment/decrements at the source language level are not done atomically, that is, the assembly language code:

```
LD R0,X /* load R0 from memory location x */
INCR R0 /* increment R0 */
STO R0,X /* store the incremented value back in X */
```

2. [10 points] What is the difference between binary and general semaphores?

A semaphore is owned or not owned (boolean). It can be tested when not owned. While owned it can be updated. It might be a simple boolean or a counting semaphore depending on the instructor and goals.
The hardware has a lot to say in how this is done so well tested system libraries are essential for getting this correct.

3. [10 points] What is a monitor?

a synchronization construct that allows threads to have both mutual exclusion and the ability to wait (block) for a certain condition to become false.

4. [20 points] What operations can be performed on a semaphore?

Worker processes can wait() or signal() a semaphore.