# 1

## 1.1 a

any $x <= 50$ will result in a speedup $>= 1$

## 1.2 b

$E = S/N$
$N = 2$
$E = 0.7$
$S = E * N$
$S = 2 * 0.7$
$S = 1.4$

## 1.3 c

$S = T_{old}/Tnew$ $S = 1000/(500+(100+10*(10))$ $S = 1000/(500+(100+100)$ $S = 1000/700$ $S = 10/7$

## 1.4 d

# 2

## 2.1 a

```
myrank =1
lock M2

CONTEXT SWITCH
myrank =2
lock M1
wait for M2

CONTEXT SWITCH
wait for M1
```

both threads would now be waiting for a lock the other holds, with no way to unlock.

### 2.2 b

yes! each would still be waiting for the other to finish what it is doing, without the other being able to finish.

### 2.3 c

A simple solution would be whenever both MUTEX may be needed, always try to acquire the same one first (like M1) before trying the other (M2)

## 3

### 3.1 d

The times decreased with the number of threads, but busy waiting still timed out. (works better with a smaller n)

### 3.2 e

needing to protected the approx value slows execution