VE482 Homework 4

Due: Oct.25
Name: Wu Qinhang
ID: 518370910041
Email: william_wu@sjtu.edu.cn

Ex1

1.1

While some thread is executing in the run-time system, directly applying a clock interrupt may result in some unexpected behavior when a certain thread is about to being blocked or unblocked. Solution: check the queue before applying the clock interrupt. Only apply the clock interrupt when no threads are switching the state.

1.2

Yes. `select()` system call is able to monitor file descriptors and tell whether it is ready for I/O operations. If it is not available for a certain OS, the alarm clock may be used by the thread to check the status before executing a system call. If the system call is blocked, the control is returned back to the thread. Such mechanism reduce the efficiency, but works.

Ex2 Monitors

Though `waituntil` can complete the same task, it will cost much more resources. Since `waituntil` has to check the value every time whenever any variable changes, the evaluation taken by it will cost much more resources than the combination of `wait` and `signal` because the latter solution only awaken the process with the signal.

Ex3 Race Condition in Bash

From my own observation, the race condition happens after the number `215`.

the original shell code:

```
#!/bin/bash
FILE=./ex3.out
# CNT=100
if ! test -f "$FILE"; then
    echo "0" >> $FILE
fi
for i in {1..20}
do
    operand=$(tail -n 1 $FILE)
    ((operand++))
    echo $operand >> $FILE
done
```

the driver program:

```
#!/bin/bash
for i in {1..20}
do
    ./ex3.sh; ./ex3.sh&
    # ./ex3_fix.sh; ./ex3_fix.sh&
done
```

the modified shell code:

```
#!/bin/bash
FILE=./ex3.out
if ! test -f "$FILE";
then
    echo "0" >> $FILE
fi
for i in {1..20}
do
(
    flock -n -x 33
    if [ $? -eq 1 ];
    then
        exit;
    fi
    operand=$(tail -n 1 $FILE)
    ((operand++))
    echo $operand >> $FILE
) 33>>$FILE
done
```

the new driver program:

```
#!/bin/bash
for i in {1..20}
do
    # ./ex3.sh; ./ex3.sh&
    ./ex3_fix.sh; ./ex3_fix.sh&
done
```

see also `./*.sh` for the programs.

Ex4 semaphore

see `./cthread.c` for more detail.

Note: use the command `clang cthread.c -o ex4 -lpthread` to compile the program.