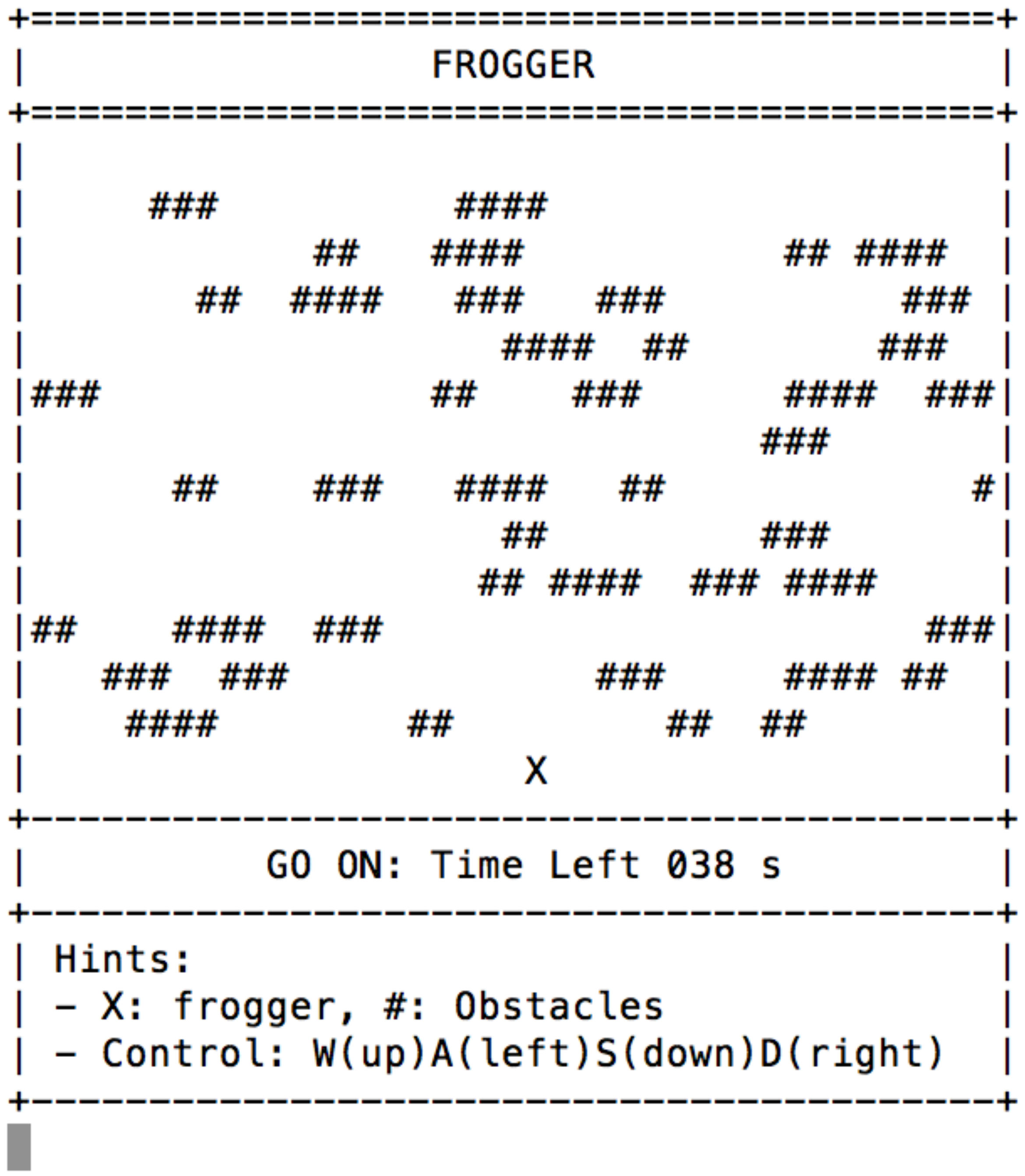
Introduction to Parallel Computing
Problem Assignment #3

Frogger

Presentation & Demo

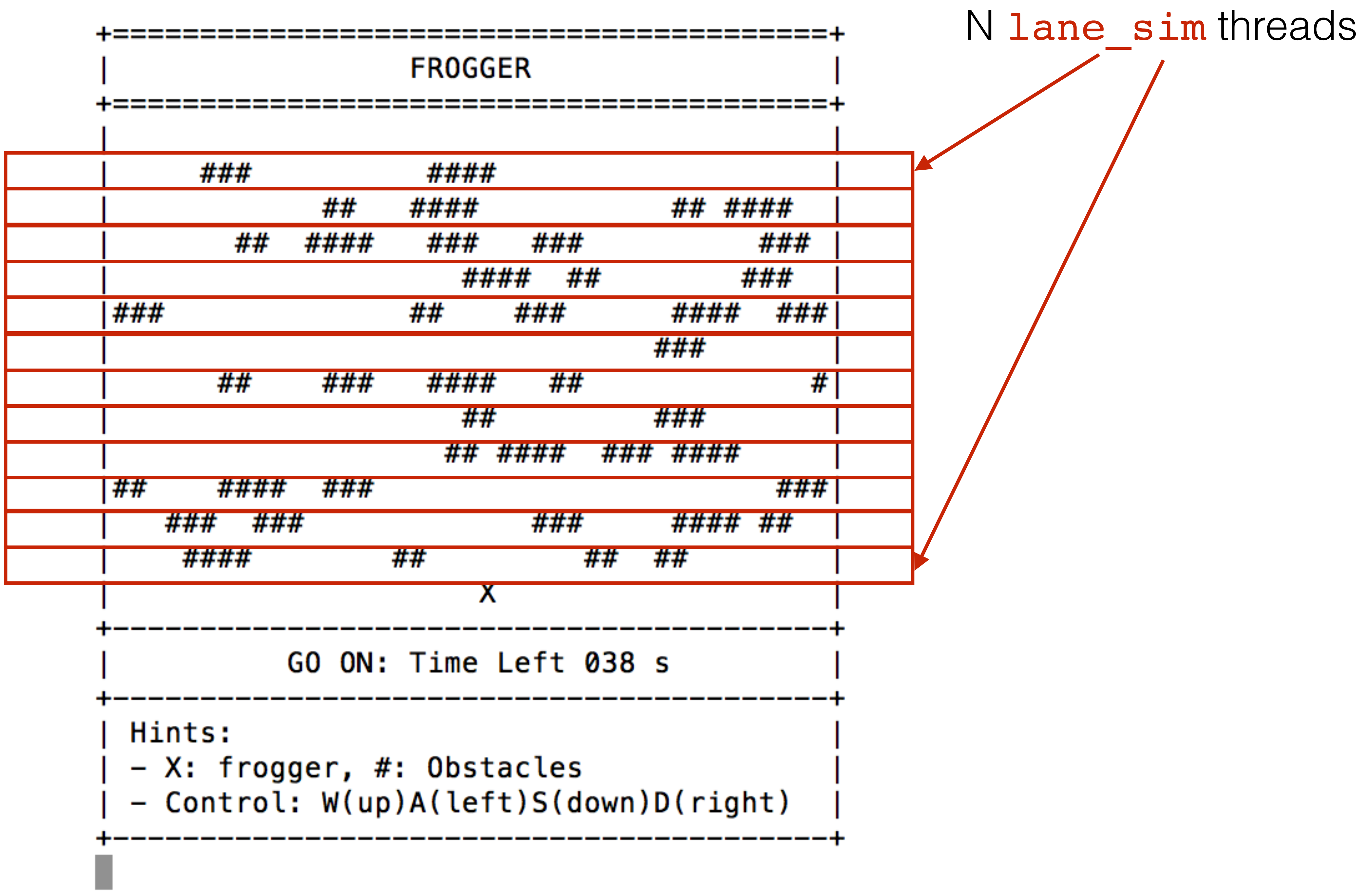
Yihong Gu
Tsinghua University
yihong15.math@gmail.com

Summary



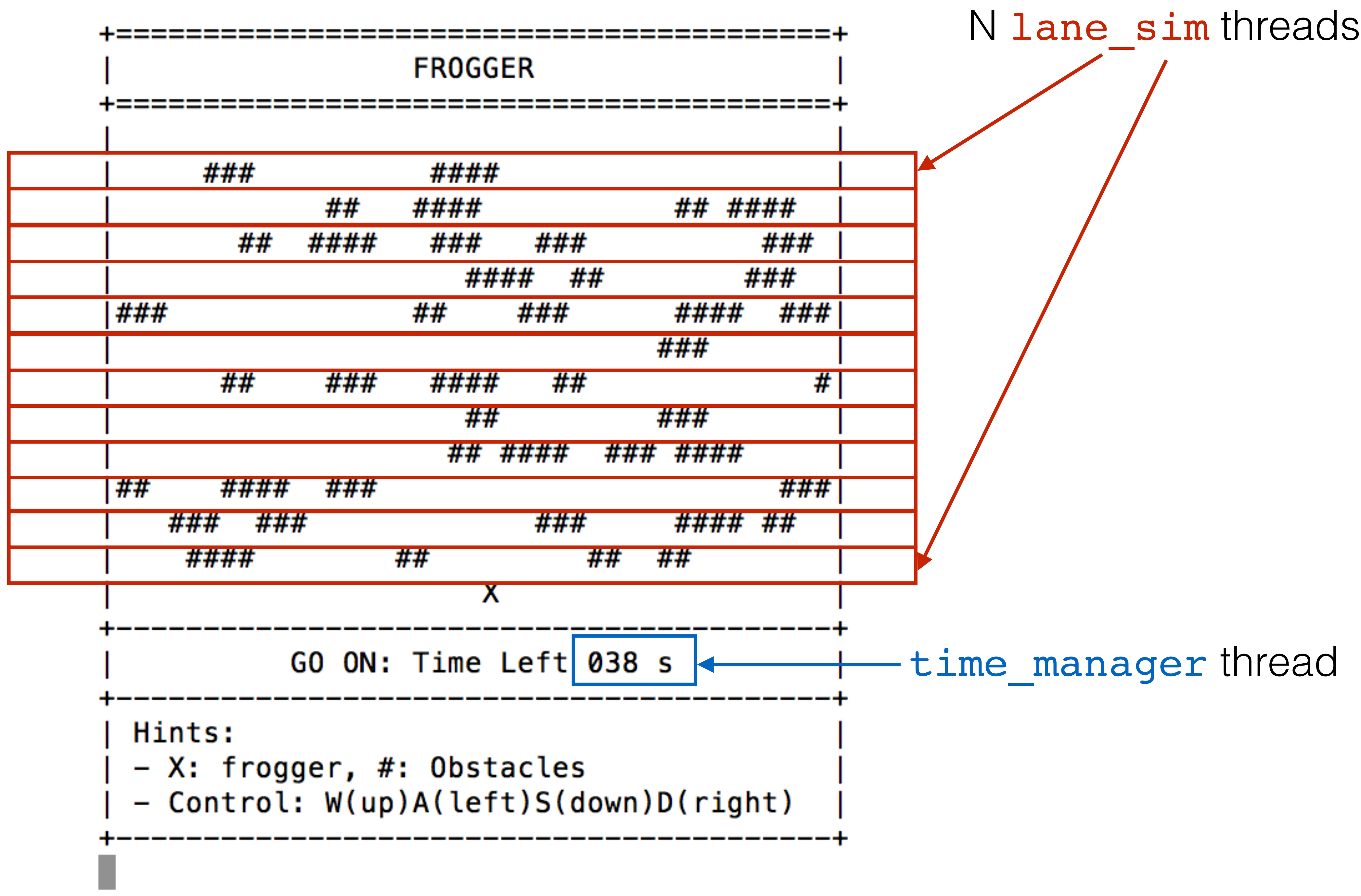
```
+=========================================+
|                 FROGGER                 |
+=========================================+
|                                         |
|     ###          ####                   |
|            ##   ####           ## ####  |
|       ##  ####   ###   ###          ### |
|                    ####  ##        ###  |
|###              ##    ###      ####  ###|
|                               ###       |
|      ##    ###   ####   ##             #|
|                    ##         ###       |
|                   ## ####  ### ####     |
|##    ####  ###                       ###|
|   ###  ###             ###     #### ##  |
|    ####        ##         ##  ##        |
|                     X                   |
+-----------------------------------------+
|          GO ON: Time Left 038 s         |
+-----------------------------------------+
| Hints:                                  |
| - X: frogger, #: Obstacles              |
| - Control: W(up)A(left)S(down)D(right)  |
+-----------------------------------------+
```

Summary



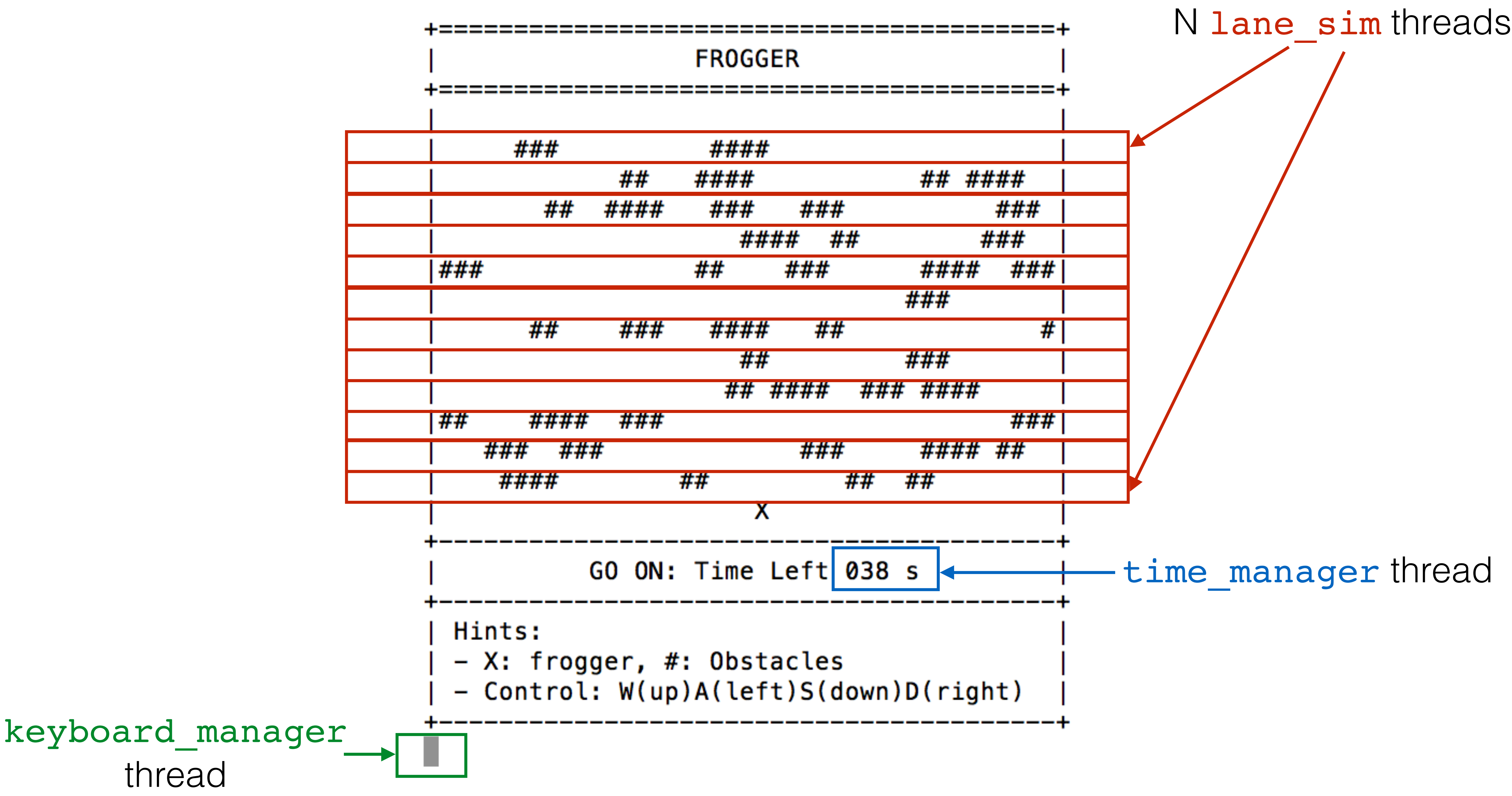
```
+=========================================+
|                 FROGGER                 |
+=========================================+
|                                         |
|     ###          ####                   |
|            ##   ####           ## ####  |
|       ##  ####   ###   ###          ### |
|                    ####  ##        ###  |
|###              ##    ###      ####  ###|
|                               ###       |
|      ##    ###   ####   ##             #|
|                    ##         ###       |
|                   ## ####  ### ####     |
|##    ####  ###                       ###|
|   ###  ###             ###     #### ##  |
|    ####        ##         ##  ##        |
|                     X                   |
+-----------------------------------------+
|          GO ON: Time Left 038 s         |
+-----------------------------------------+
| Hints:                                  |
| - X: frogger, #: Obstacles              |
| - Control: W(up)A(left)S(down)D(right)  |
+-----------------------------------------+
```

Summary



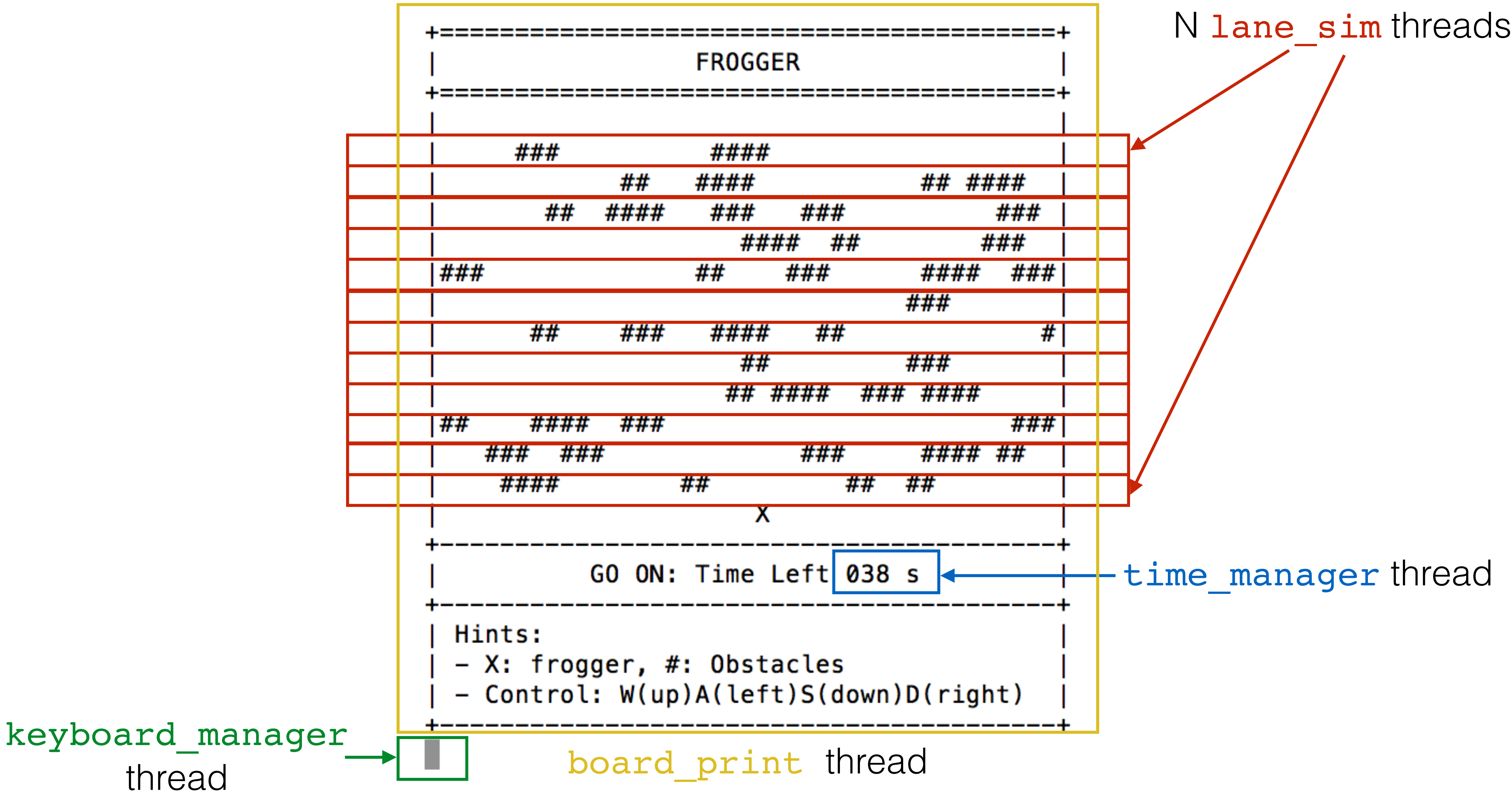
```
+=========================================+
|                 FROGGER                 |
+=========================================+
|                                         |
|     ###          ####                   |
|            ##   ####           ## ####  |
|       ##  ####   ###   ###          ### |
|                    ####  ##        ###  |
|###              ##    ###      ####  ###|
|                               ###       |
|      ##    ###   ####   ##             #|
|                    ##         ###       |
|                   ## ####  ### ####     |
|##    ####  ###                       ###|
|   ###  ###             ###     #### ##  |
|    ####        ##         ##  ##        |
|                     X                   |
+-----------------------------------------+
|          GO ON: Time Left 038 s         |
+-----------------------------------------+
| Hints:                                  |
| - X: frogger, #: Obstacles              |
| - Control: W(up)A(left)S(down)D(right)  |
+-----------------------------------------+
```

Summary



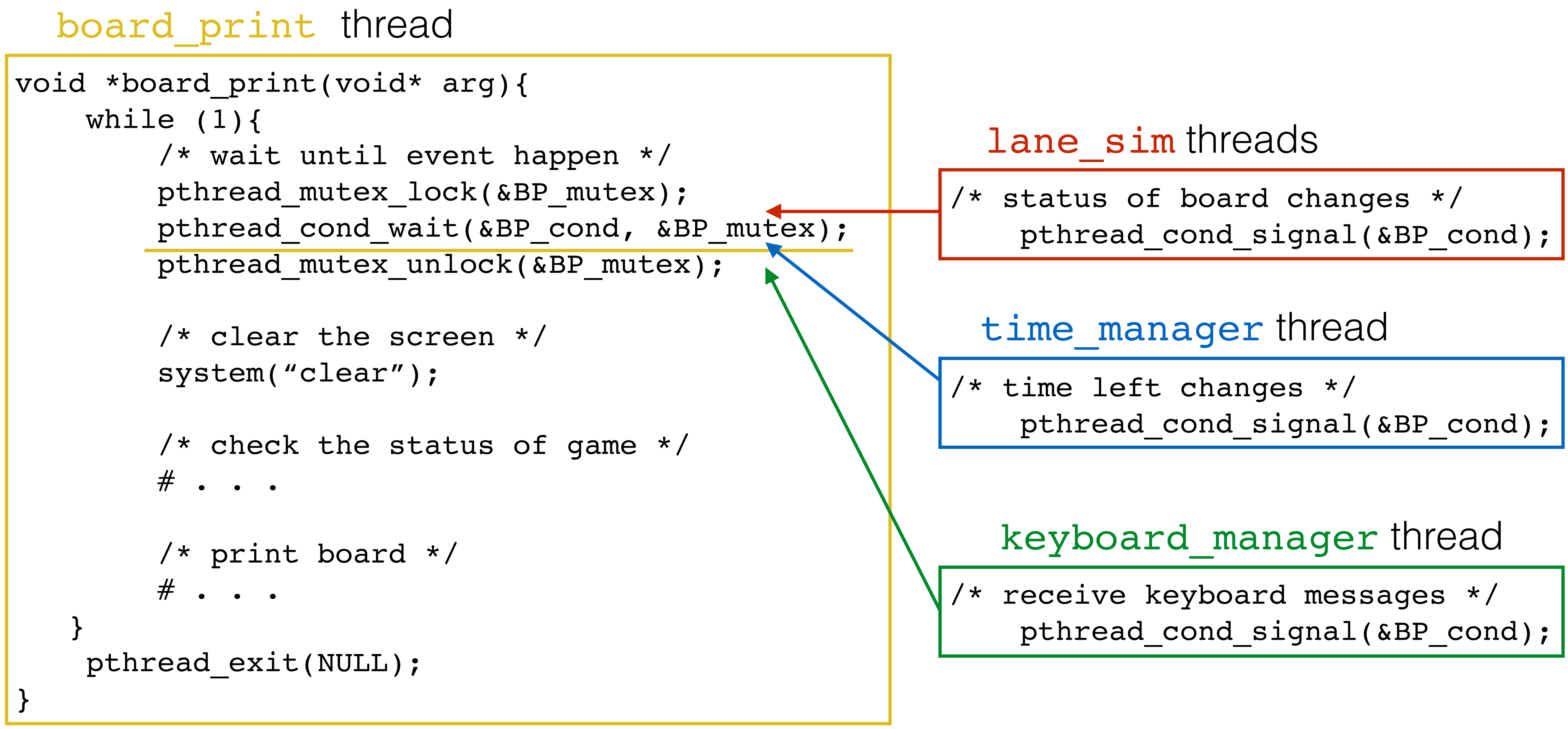
```
+=========================================+
|                 FROGGER                 |
+=========================================+
|                                         |
|     ###          ####                   |
|            ##   ####           ## ####  |
|       ##  ####   ###   ###          ### |
|                    ####  ##        ###  |
|###              ##    ###      ####  ###|
|                               ###       |
|      ##    ###   ####   ##             #|
|                    ##         ###       |
|                   ## ####  ### ####     |
|##    ####  ###                       ###|
|   ###  ###             ###     #### ##  |
|    ####        ##         ##  ##        |
|                     X                   |
+-----------------------------------------+
|          GO ON: Time Left 038 s         |
+-----------------------------------------+
| Hints:                                  |
| - X: frogger, #: Obstacles              |
| - Control: W(up)A(left)S(down)D(right)  |
+-----------------------------------------+
```

Summary



```
+=========================================+
|                 FROGGER                 |
+=========================================+
|                                         |
|     ###          ####                   |
|            ##   ####           ## ####  |
|       ##  ####   ###   ###          ### |
|                    ####  ##        ###  |
|###              ##    ###      ####  ###|
|                               ###       |
|      ##    ###   ####   ##             #|
|                    ##         ###       |
|                   ## ####  ### ####     |
|##    ####  ###                       ###|
|   ###  ###             ###     #### ##  |
|    ####        ##         ##  ##        |
|                     X                   |
+-----------------------------------------+
|          GO ON: Time Left 038 s         |
+-----------------------------------------+
| Hints:                                  |
| - X: frogger, #: Obstacles              |
| - Control: W(up)A(left)S(down)D(right)  |
+-----------------------------------------+
```

board_print thread



board_print thread

```
void *board_print(void* arg){
    while (1){
        /* wait until event happen */
        pthread_mutex_lock(&BP_mutex);
        pthread_cond_wait(&BP_cond, &BP_mutex);
        pthread_mutex_unlock(&BP_mutex);

        /* clear the screen */
        system(“clear”);

        /* check the status of game */
        # . . .

        /* print board */
        # . . .
   }
    pthread_exit(NULL);
}
```

lane_sim threads

```
/* status of board changes */
   pthread_cond_signal(&BP_cond);
```

time_manager thread

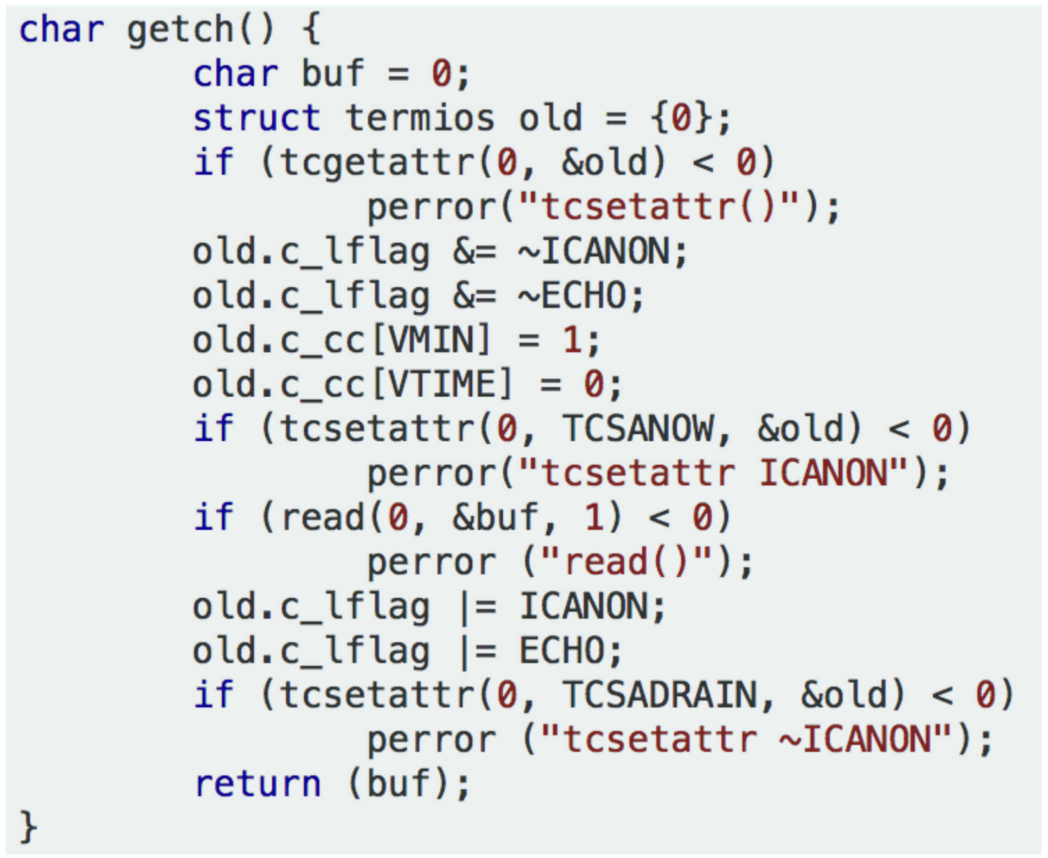
```
/* time left changes */
   pthread_cond_signal(&BP_cond);
```

keyboard_manager thread

```
/* receive keyboard messages */
   pthread_cond_signal(&BP_cond);
```

`keyboard_manager` thread

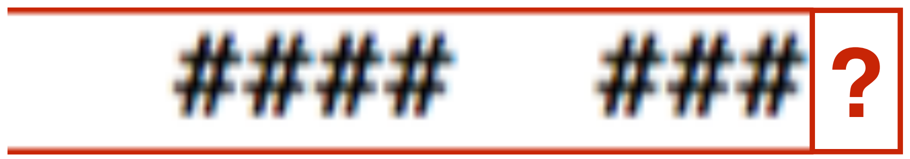
- Kernel Problem: capture characters from standard input without waiting for enter to be pressed
  - getch() in Windows
  - using unistd.h in Linux [1]



```c
char getch() {
        char buf = 0;
        struct termios old = {0};
        if (tcgetattr(0, &old) < 0)
                perror("tcsetattr()");
        old.c_lflag &= ~ICANON;
        old.c_lflag &= ~ECHO;
        old.c_cc[VMIN] = 1;
        old.c_cc[VTIME] = 0;
        if (tcsetattr(0, TCSANOW, &old) < 0)
                perror("tcsetattr ICANON");
        if (read(0, &buf, 1) < 0)
                perror ("read()");
        old.c_lflag |= ICANON;
        old.c_lflag |= ECHO;
        if (tcsetattr(0, TCSADRAIN, &old) < 0)
                perror ("tcsetattr ~ICANON");
        return (buf);
}
```

[1] StackOverflow https://stackoverflow.com/questions/421860/capture-characters-from-standard-input-without-waiting-for-enter-to-be-pressed

lane_sim thread



- Simulation Method (Markov Chain)
  - block unit (empty or obstacle)
  - since now we have L continuous obstacles, the probability that the next block is obstacle is prob[L]
    - prob[0] = p
    - prob[1] = 1.0, prob[2] = 0.7, prob[3] = 0.4, prob[4] = 0.0
  - Every T microsecond add a new block and send a signal
    - T ~ Uniform(BASE, BASE+RANGE)
- N `lane_sim` threads and `board_print` thread share board memory (no mutex)

`lane_sim` thread (cont)

- since now we have L continuous obstacles, the probability that the next block is obstacle is prob[L]
  - prob[0] = p
- Every T microsecond add a new block and send a signal
  - T ~ Uniform(BASE, BASE+RANGE)
- Difficulty
  - easy(0), medium(1), hard(2)
  - Higher difficulty level means
    - smaller BASE, RANGE
    - larger ‘p’

Thank You

Q&A