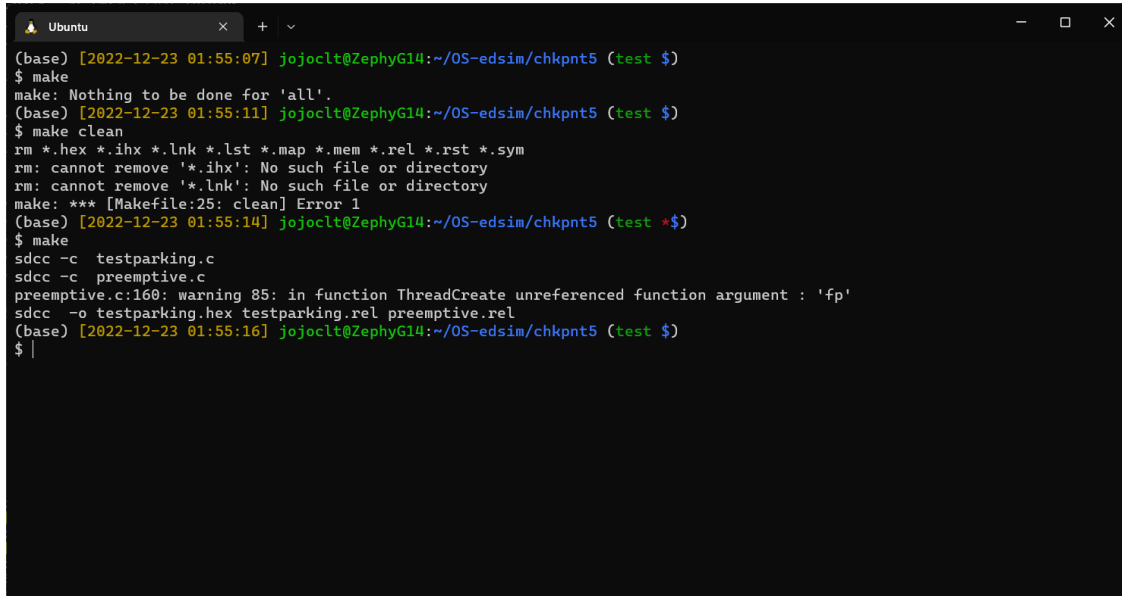


Programming Project Checkpoint#5

姚林飛 109006243

[1] Typescript for compilation



```
(base) [2022-12-23 01:55:07] jojoelt@ZephyG14:~/OS-edsim/chkpnt5 (test $)
$ make
make: Nothing to be done for 'all'.
(base) [2022-12-23 01:55:11] jojoelt@ZephyG14:~/OS-edsim/chkpnt5 (test $)
$ make clean
rm *.hex *.ihx *.lnk *.lst *.map *.mem *.rel *.rst *.sym
rm: cannot remove '*.ihx': No such file or directory
rm: cannot remove '*.lnk': No such file or directory
make: *** [Makefile:25: clean] Error 1
(base) [2022-12-23 01:55:14] jojoelt@ZephyG14:~/OS-edsim/chkpnt5 (test *$)
$ make
sdcc -c testparking.c
sdcc -c preemptive.c
preemptive.c:160: warning 85: in function ThreadCreate unreferenced function argument : 'fp'
sdcc -o testparking.hex testparking.rel preemptive.rel
(base) [2022-12-23 01:55:16] jojoelt@ZephyG14:~/OS-edsim/chkpnt5 (test $)
$ |
```

[2] Screenshots and explanation

1) delay(n), now()

Delay implementation - When it is called, the thread is delayed by n time units (current_time + n), where the time units are based on the timer-0 ISR.

When the timer-0 ISR is called, it will increase the global timer when the thread is at thread#0.

If all the threads call delay() and finish their delays all at the same time, it will use the first found thread that is free.

Note: Everytime Thread is Exit, it will go through all threads (car) again, to check whether there is a car that wants to go in, but not yet knows that the lot is free.

2) Thread Termination and Creation

I added the Semaphores for Thread Creation and Termination.

```
1 ThreadID ThreadCreate(FunctionPtr fp) {
2     SemaphoreWait(threads);
3     EA = 0;
4
5     for (createdThread = 0; createdThread < MAXTHREADS;
6         createdThread++) {
7         tmp = bitmap & (1 << createdThread);
8         if (tmp) continue;
9         break;
10    }
```

```
1 void ThreadExit(void) {
2     __critical {
3         bitmap = bitmap ^ (1 << currentThread);
4         SemaphoreSignal(threads);
5     }
6     EA = 0;
7     currentThread = 0;
```

3. Parking Lot Example

Data Memory															
	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E
00	00	00	01	00	00	01	00	00	08	00	00	00	00	0F	0D
10	00	00	00	00	00	05	01	01	00	00	00	00	00	0D	05
20	47	57	67	77	00	00	02	5F	41	00	00	1A	04	02	20
30	FF	04	06	05	00	00	00	02	03	04	02	03	05	04	06
40	01	20	06	E7	07	0A	09	00	08	0A	09	0A	09	00	00
50	00	07	0A	09	AC	02	C8	00	08	0A	09	00	00	00	00
60	00	07	0A	09	69	03	D0	00	08	0A	09	00	00	00	00
70	00	07	0A	09	EB	01	D8	00	08	0A	09	00	00	00	00

Copyright ©2005-2022 James Rogers

Remove All Breakpoints

U	No Parity
C0:	0-2@0
C1:	0-3@1
C2:	2-5@0
C3:	3-4@1
C4:	4-6@1

Where 0x35-0x39 is arrival time and 0x3A to 0x3E is departure time, and 0x2B is the parking_lot status by bit.