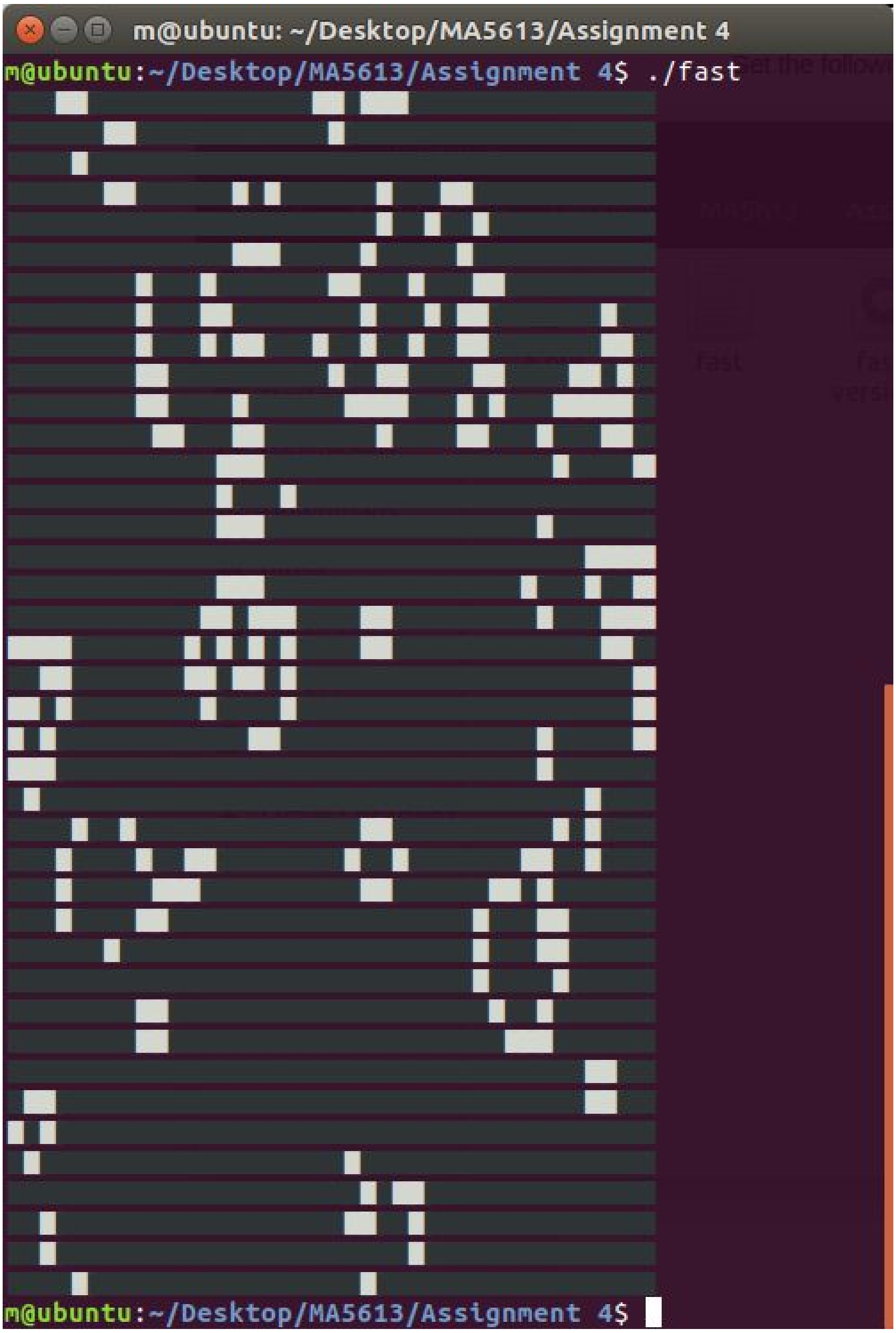

```
m@ubuntu:~/Desktop/MA5613/Assignment 4$ ./fast
m@ubuntu:~/Desktop/MA5613/Assignment 4$
```

<u>Using the #defines for control:</u>

Set the following to either 0 or 1 in the .c files to control what the program does.

PRINT_RESULTS when set will print all results to the terminal. Live cells are represented with a white square, and dead cells with a black square. This may not look good if you use it with a terminal with a white background.

SLEEP_BETWEEN_PRINTS when set will make the program sleep for 1 second in between prints. If you want the final output as soon as possible then set this to 0.

RECORD_TIME will record and print the total time taken. It is recommended that printing and sleeping between prints are disabled when using this.

Making it as fast as possible:

Out of curiosity I decided to see how fast I could get this to run.
I created a fast version of the code that uses loop unrolling, OpenMP to do work in parallel, and instead of copying memory between the grids it simply uses a pointer to swap the grids.

The original version takes ~8000 microseconds on my computer
The faster version takes ~2000 microseconds.
Loop unrolling by itself reduced the time taken to ~5500 microseconds.
I think it could be made even faster using SIMD instructions, but I didn't have time to do that.
Times were recorded with printing and sleeping disabled.

Note: When compiling the fast version please include -fopenmp as a parameter.
You may change the NUM_THREADS definition to set how many threads OpenMP will use.