```
 1 # 3-for-a-while
 2
 3 ## `stars.c`
 4 - double loops
 5 - code style:
 6   - empty lines
 7     - `#include`
 8   - [ ] file comments
 9 - `for (int i = 0) + for (int j = 0)`
10
11 ## `is-prime.c`
12 - double loops
13 - `int is_prime = 1;`: why 1? why not 0?
14 - `if (is_prime)` vs. `if (is_prime != 0)` vs. `if (
   is_prime == 1)`
15 - testing
16   - https://www.wolframalpha.com/input?i=+primes+less+than
   +100000
17   - mma: `PrimePi[100000]`
18   - `number = 2`
19 - `break`
20 - `i * i <= number` vs. `i * i < number`
21 - `stdbool.h`
22 - [x] timing
23   - `clock_t start = clock(); clock_t end = clock(); (end
    - start) / CLOCKS_PER_SEC`
24
25 # `binary-search.c`
26 - already sorted array
27   - Fib
28 - `int index = -1;`
29   - `printf`
30 - `break`
31 - testing
32   - `1`: the leftmost/rightmost one
33   - search for the leftmost/rightmost one
34 - [ ] learn from the standard library???
35   - `(low + high) / 2`
36     - `low + (high - low) / 2`
37     - [ ] try it???
38
39 ## `digits.c`
40 - testing
41 - `do-while`
```

```
42
43 ## `palindrome.c`
44 - `#define`: pre-processing
45 - `scanf("%20s", string);`
46 - `strlen`
47 - comma expression
48
49 ## `selection-sort.c`
50 - preparation: scanf
51 - with comments
52 - `swap`
53 - `while (scanf ...)`
54   - https://en.cppreference.com/w/c/io/fscanf
55     - Number of receiving arguments successfully assigned
    (which may be zero in case a matching failure occurred
   before the first receiving argument was assigned)
56     - or `EOF` if input failure occurs before the first
   receiving argument was assigned
57   - How to run this?
58     - Linux: `Ctrl + D` at the beginning of a line
59     - Mac: `Cmd + D` at the beginning of a line
60     - Windows: `Ctrl + Z` at the beginning of a line
61 - more `printf`
62 - `I/O indirection`
63   - Linux/Windows Cmd
```