# Answers

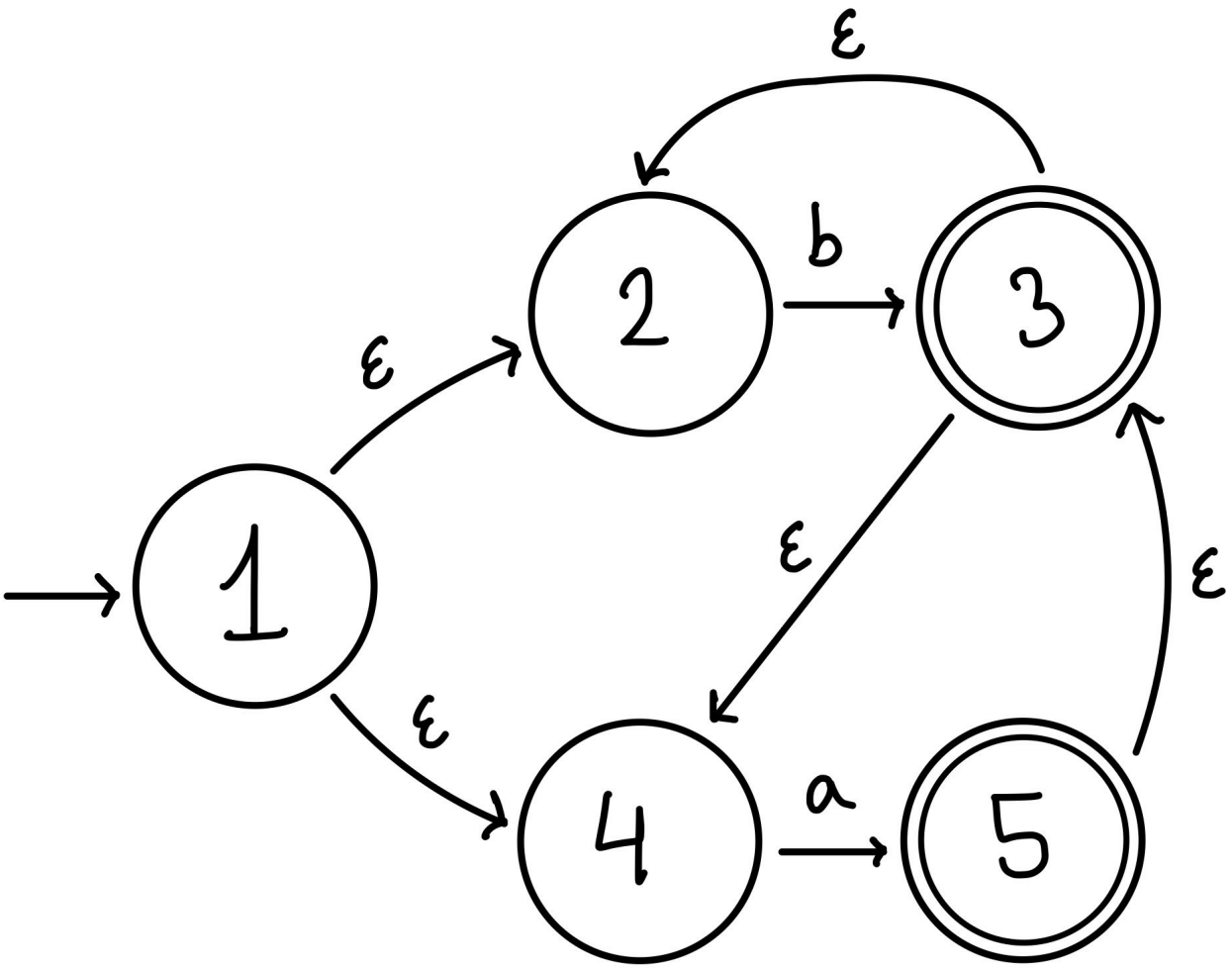
## PLC 2.4 & 2.5

Our answers to these exercises can be viewed here: https://github.com/REXKrash/PRDAT2022/blob/main/Lectures/Lec02/ex2_4Handout.fs (See ex2_4Handout.fs)

## PLC 3.2

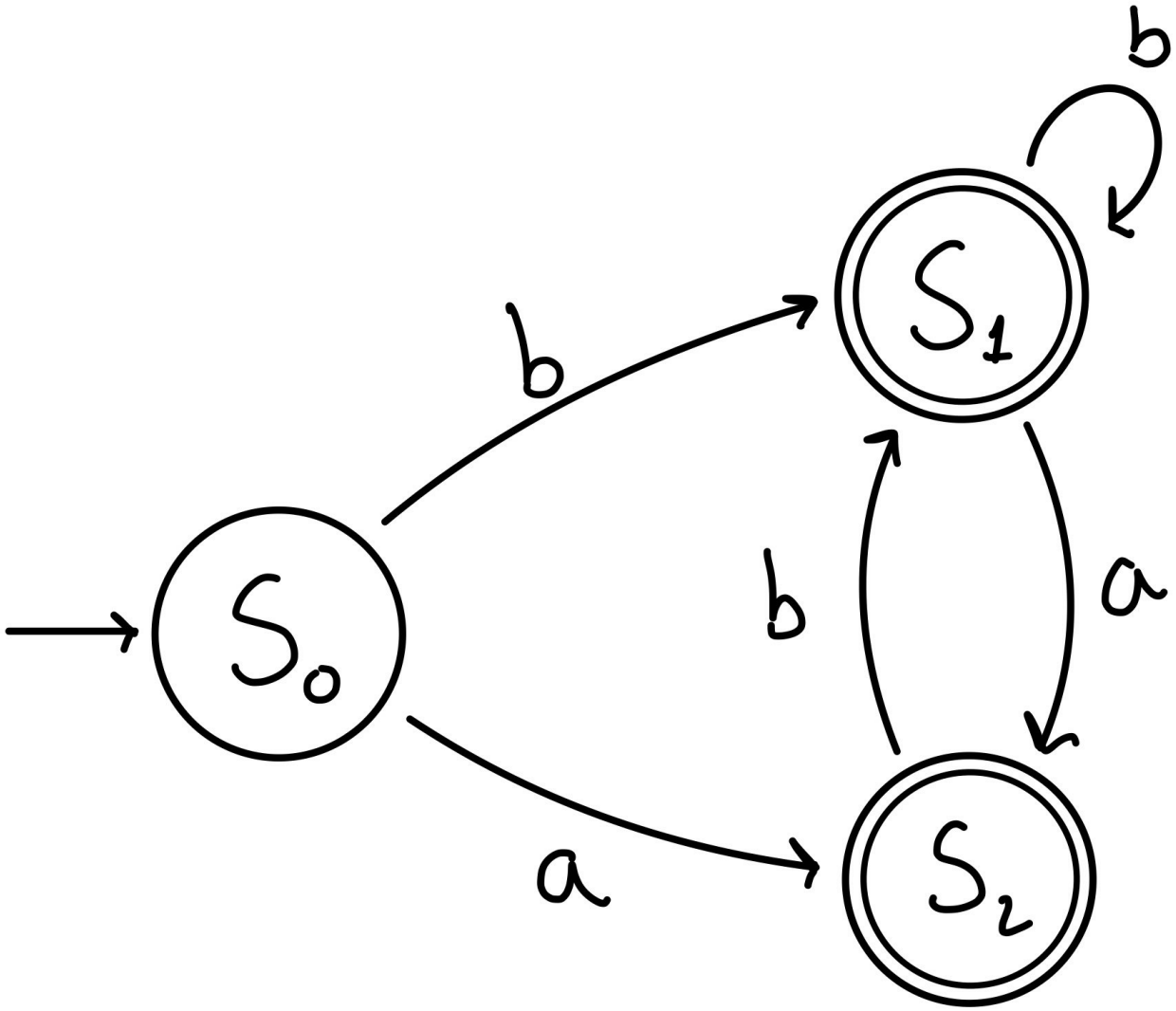
Our regular expression solution for a regular expression which recognizesallsequencesconsistingof *a* and *b* where two *a*'s are always separated by at least one *b*, is "^(blabla$)+$".

### NFA




### DFA

| DFA | At A | At B | NFA-states |
|---|---|---|---|
| S0 | S2 | S1 | {1,2,4} |
| S1 | S2 | S1 | {3} |
| S2 | X | S1 | {5} |




## BCD 2.1

Note: We were confused about the phrasing of the question, and asked a TA. For example, we ewre cofused whether "All number-strings that have the value 42." meant any number that had "42" in it, such as 1230420123, or if it is any number that equals 42, such as 0000042. We were told the former was the correct understandig, and have done the excercises in such a mannner.

**a)**

([0-9]*42[0-9]*)+

**b)**

^(?!.*42.*)[0-9]+

**c)**

0*([1-9][0-9]{2,}|4[3-9]|[5-9][0-9])

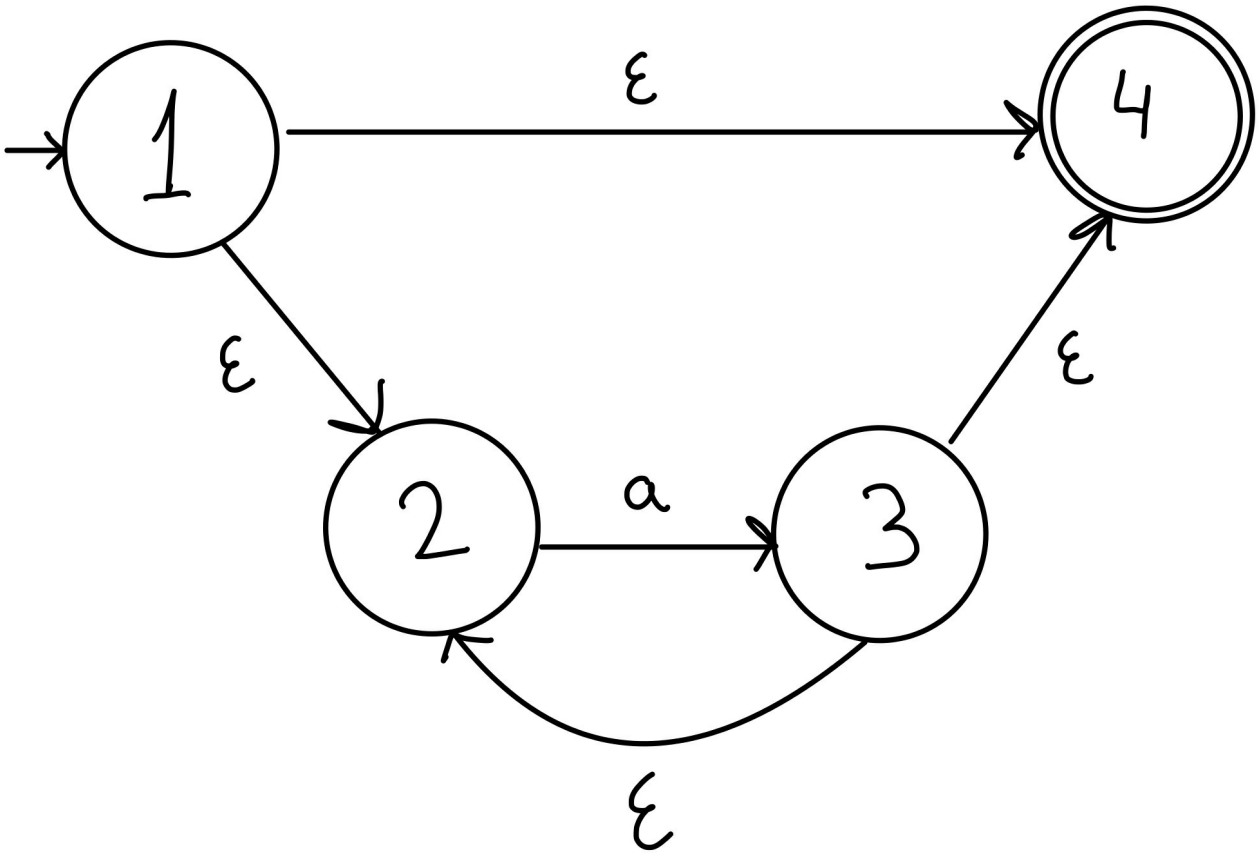
## BCD 2.2

**a)**

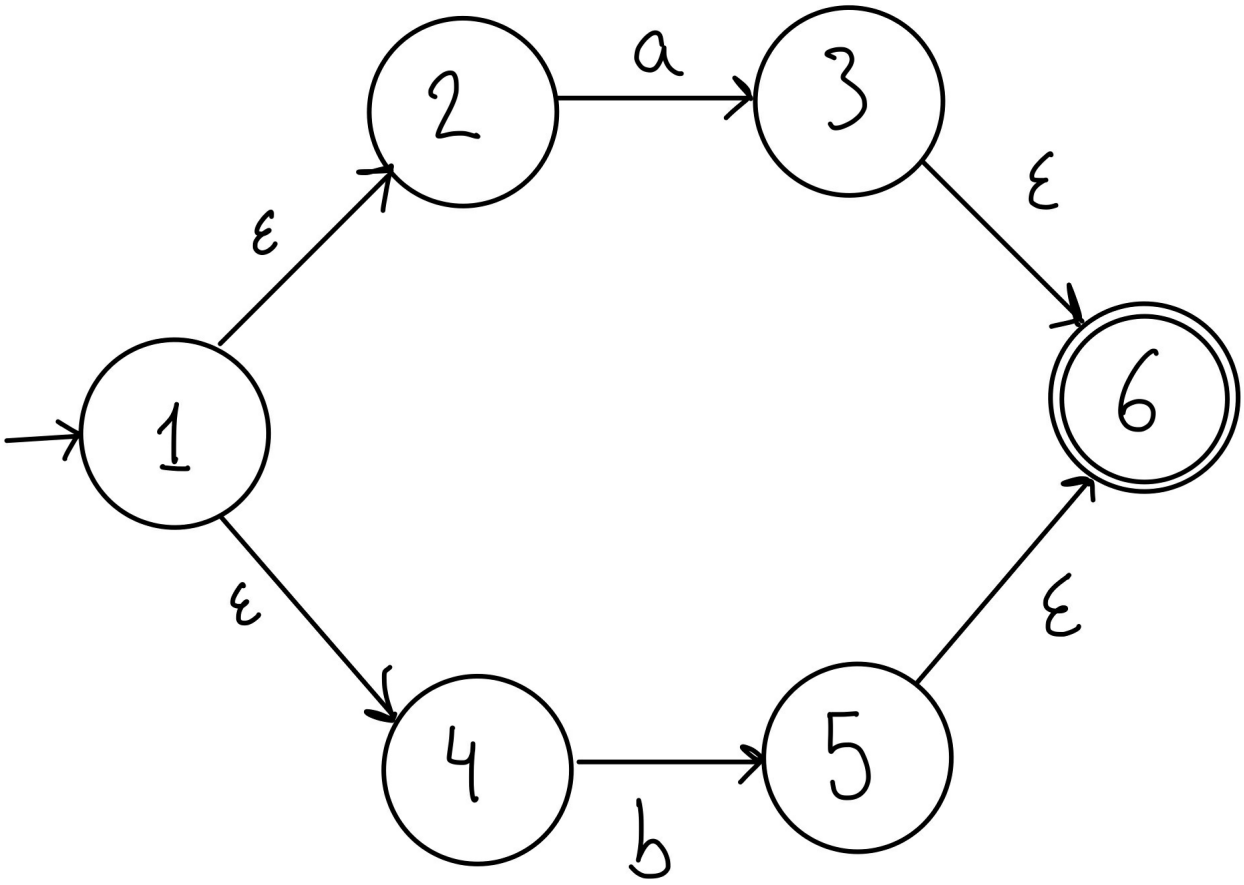
To make a NFA, we have divided the regular expression into 3 parts (a*, alb and aa):


We then make seperate NFAs for each part:

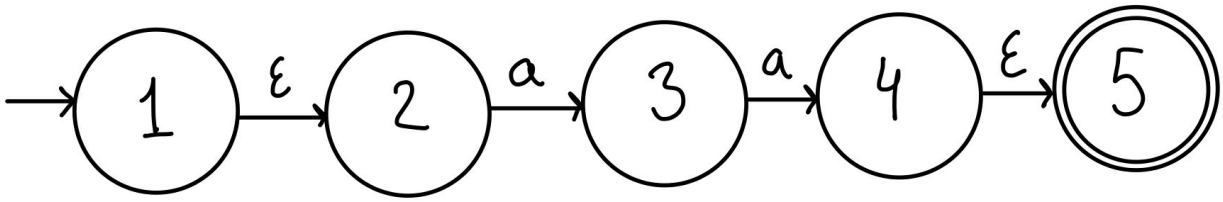
**NFA for *a**



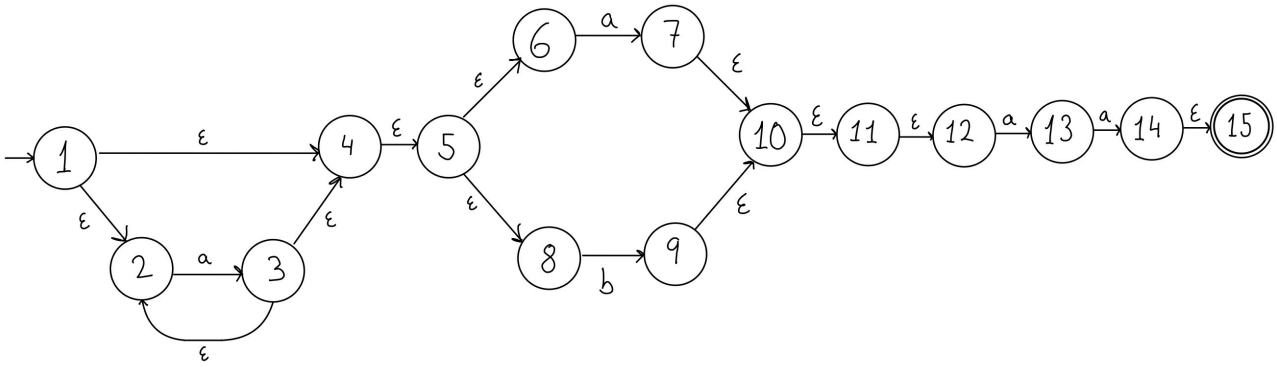

**NFA for *alb***




**NFA for *aa***




Lastly, we will unify each seperate NFA into 1 final NFA:



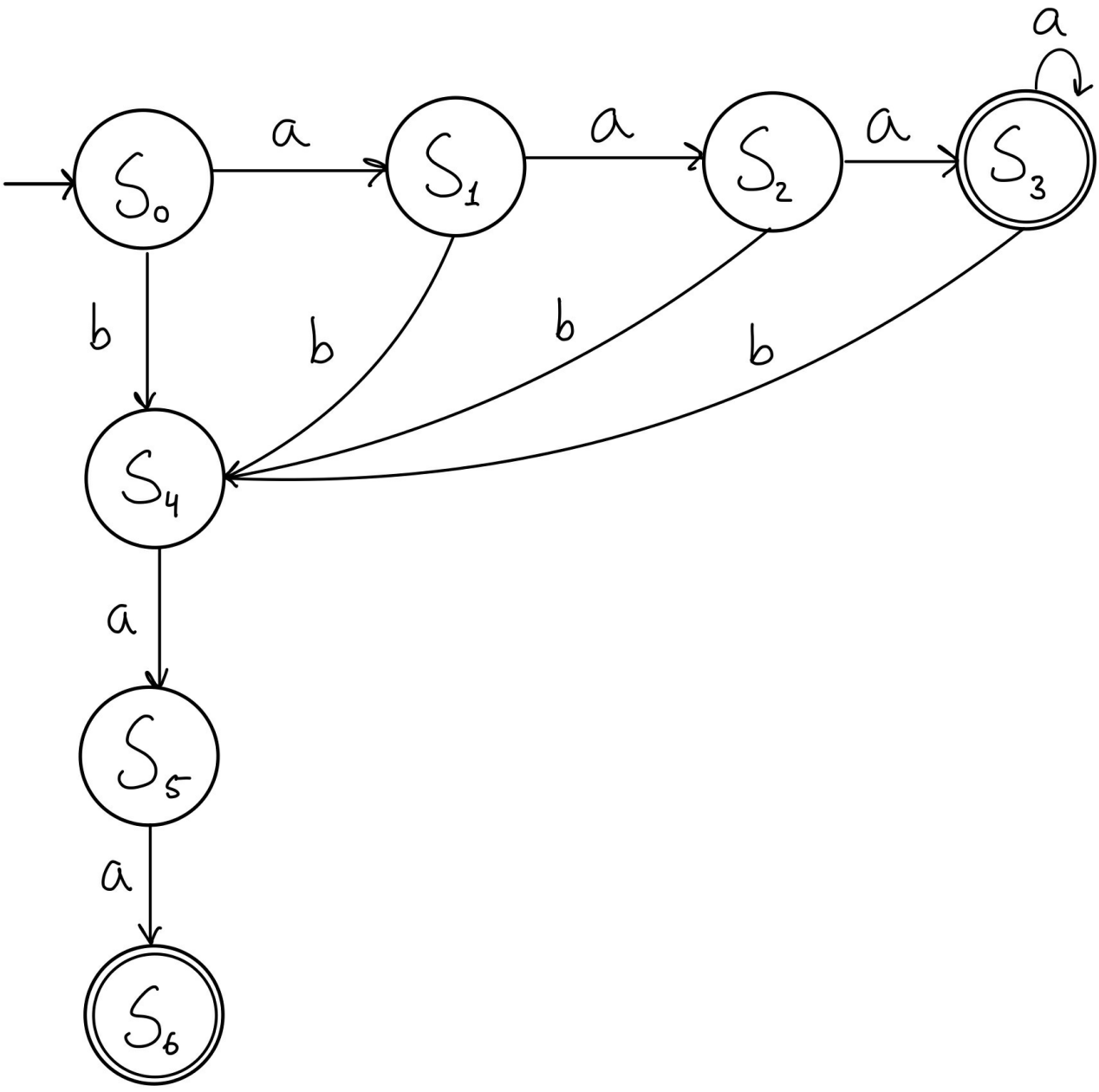

## b)

Through building this DFA, we have made the followig matrix:

| DFA At A | At B | NFA-states |
|---|---|---|

| | | | |
|---|---|---|---|
| S0 | S1 | S4 | {1,2,4,5,6,8} |
| S1 | S2 | S4 | {2,3,4,5,6,7,8,10,11,12} |
| S2 | S3 | S4 | {2,3,4,5,6,7,8,10,11,12,13} |
| S3 | S3 | S4 | {2,3,4,5,6,7,8,10,11,12,13,14,15} |
| S4 | S5 | X | {9,10,11,12} |
| S5 | S6 | X | {13} |
| S6 | X | X | {14,15} |

This translates into the followig DFA:




## HelloLex exercises:

### Question 1:

What are the regular expressions involved, and which semantic values are they associated with?

[0-9]

This is a single digit which is one of the natural numbers

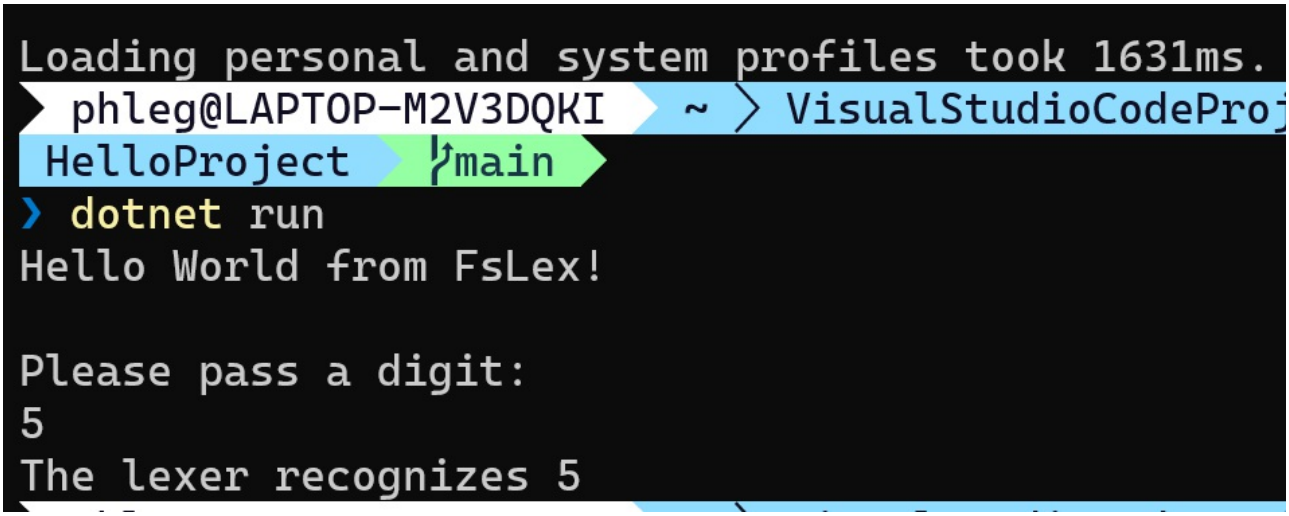
### Question 2:

Which additional file is generated during the process?
hello.fs

How many states are there by the automaton of the lexer?
3 states

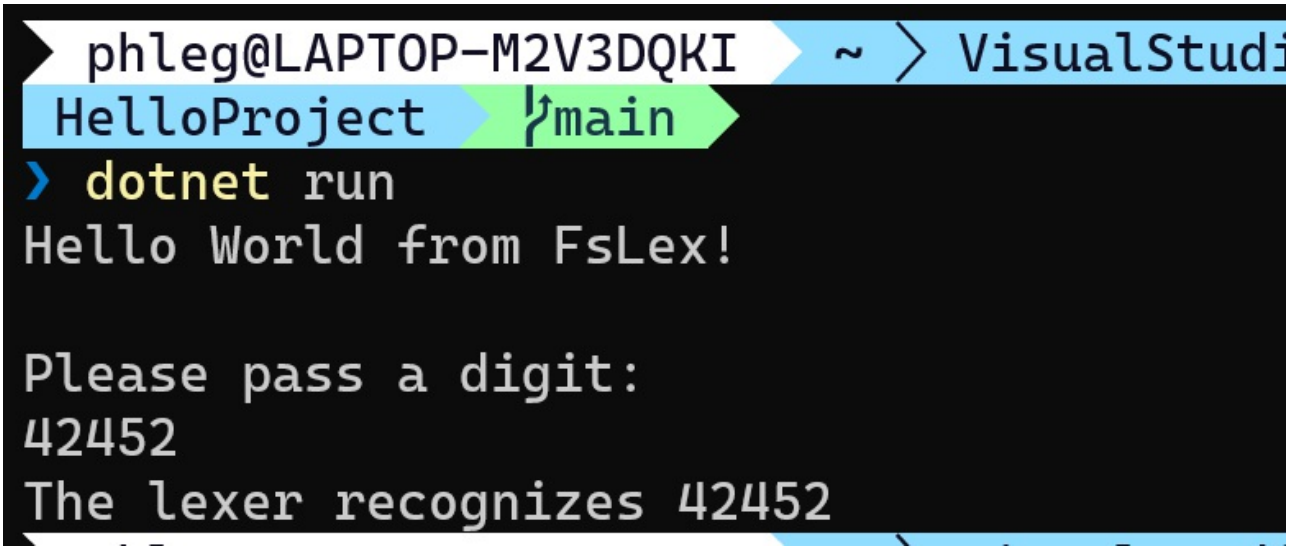
### Question 3:

Compile and run the generated program




### Question 4:

Extend the lexer specification hello.fsl to recognize numbers of more than one digit. New lexer specification is hello2.fsl. Generate hello2.fs, compile and run the generated program.



**Question 5:**

Extend the lexer specification hello2.fsl to recognize floating numbers. New lexer specification is hello3.fsl. Generate hello3.fs, compile and run the generated program.