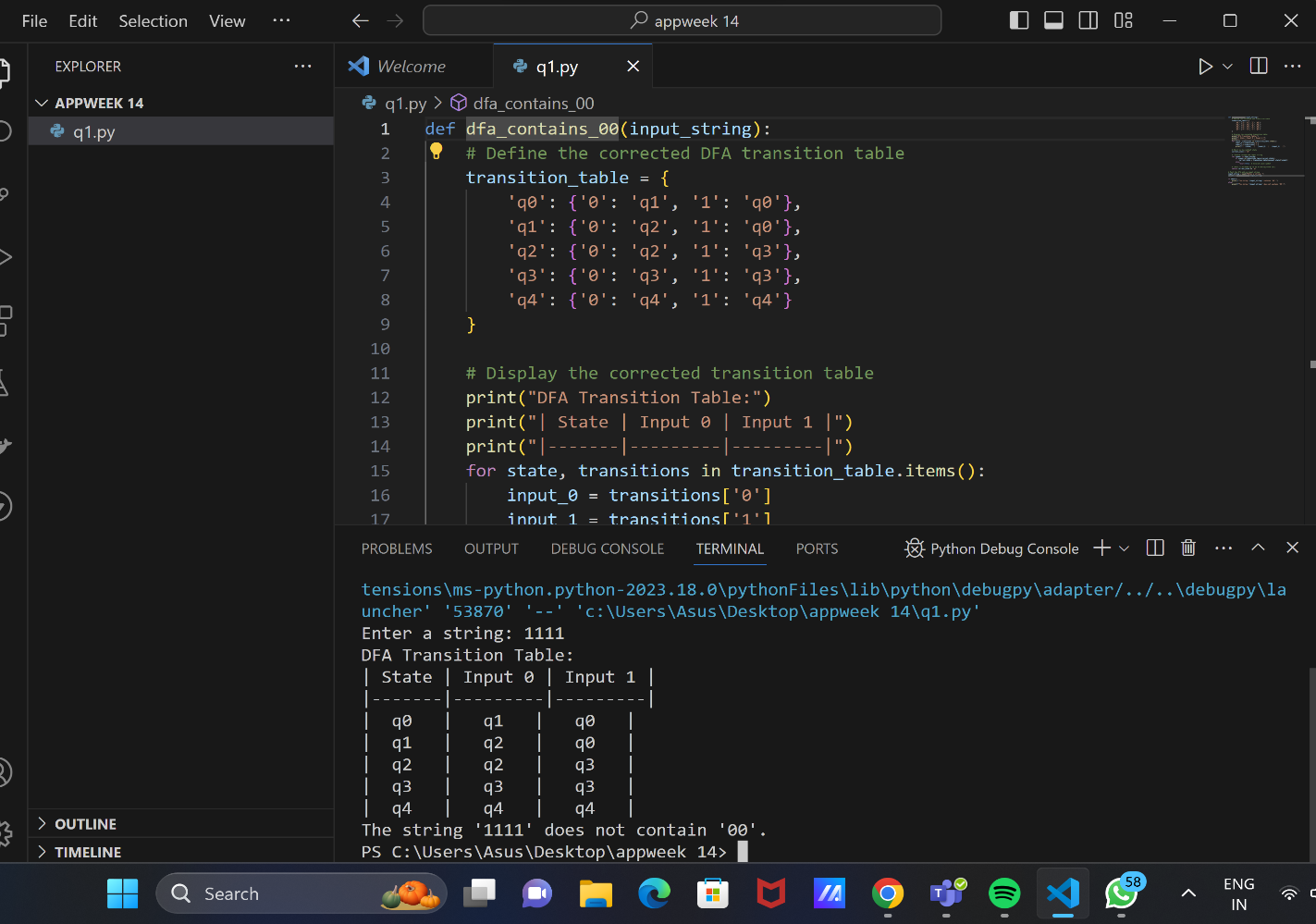
**App week 14**

**Automata Programming Paradigm**

**DFA**

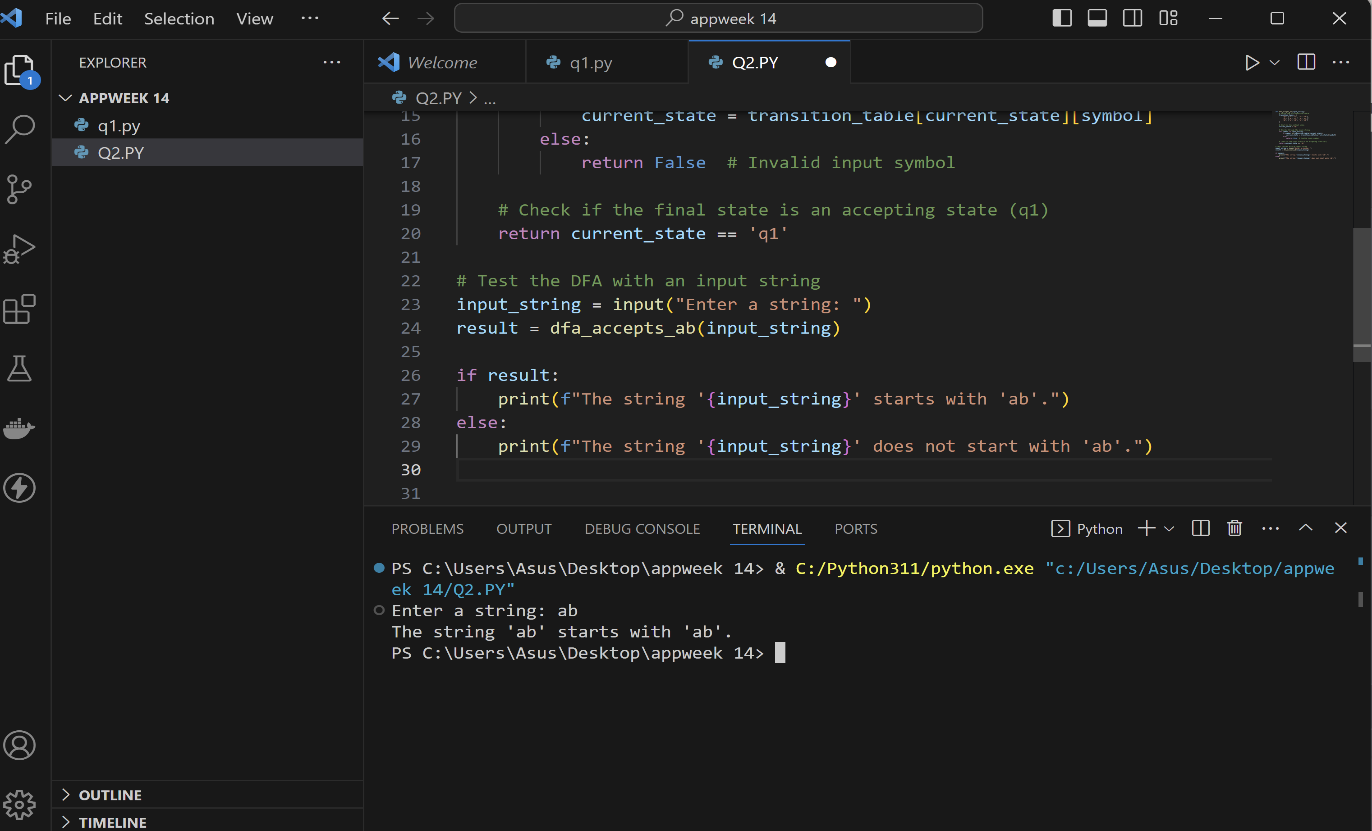
1. Draw a DFA and give the transition table for the language that accepts all and

only those strings that contain 00

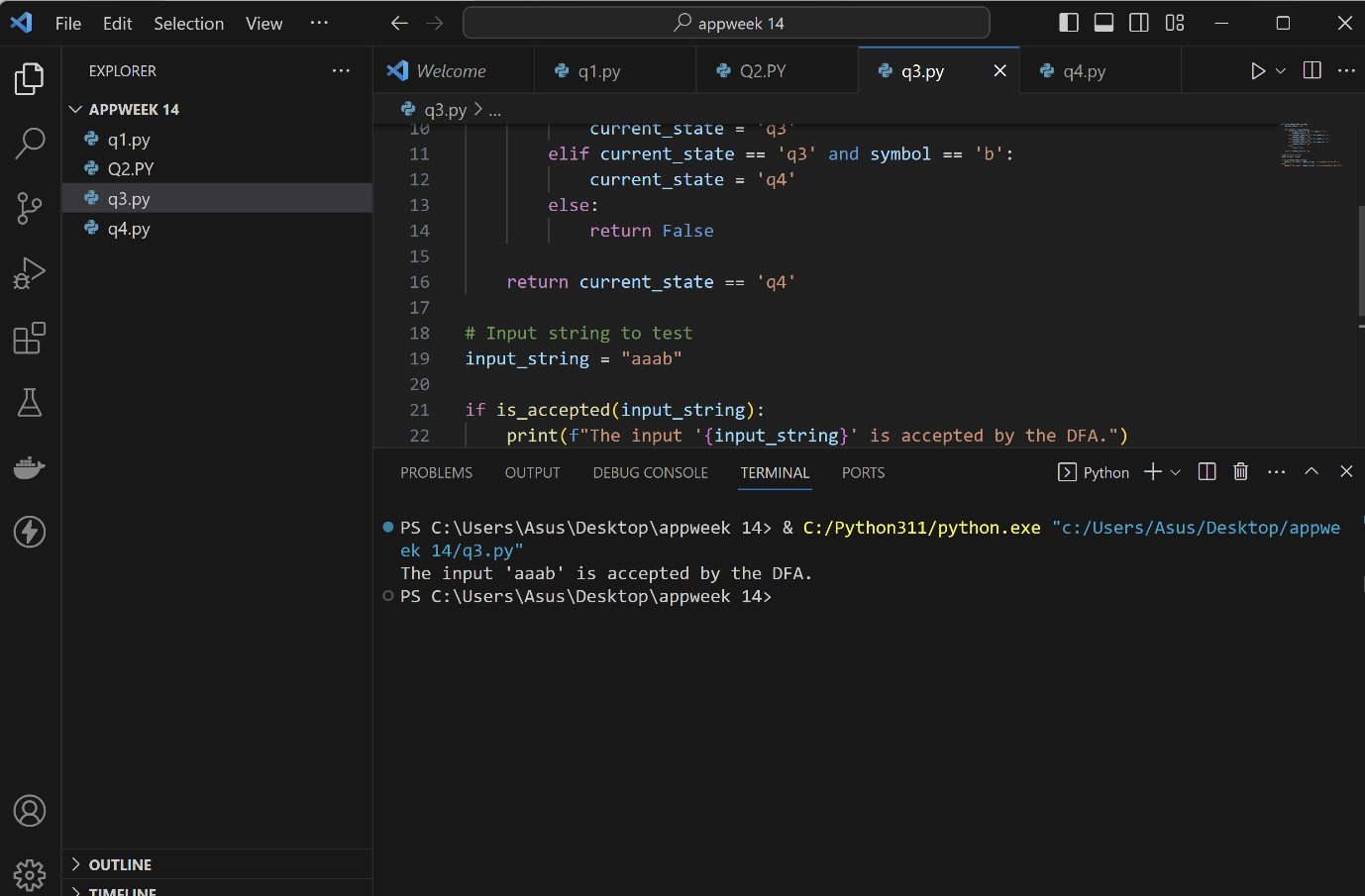


2. Draw a DFA and give the transition table for the language accepting strings

starting with ‘ab’ over input alphabets ∑ = {a, b}



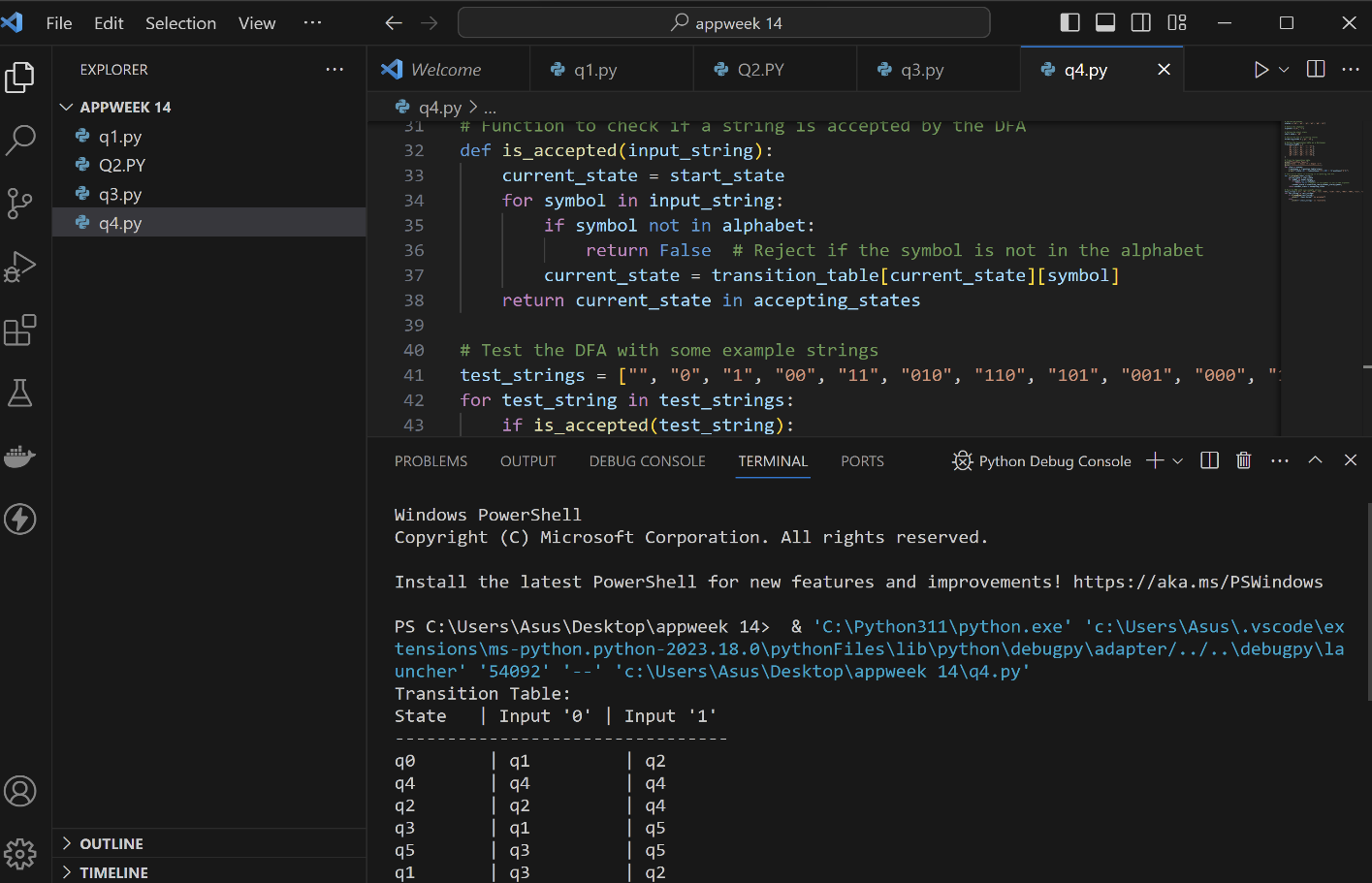
3. Construct a DFA with ∑ = {a, b} accepts the only input “aaab”.

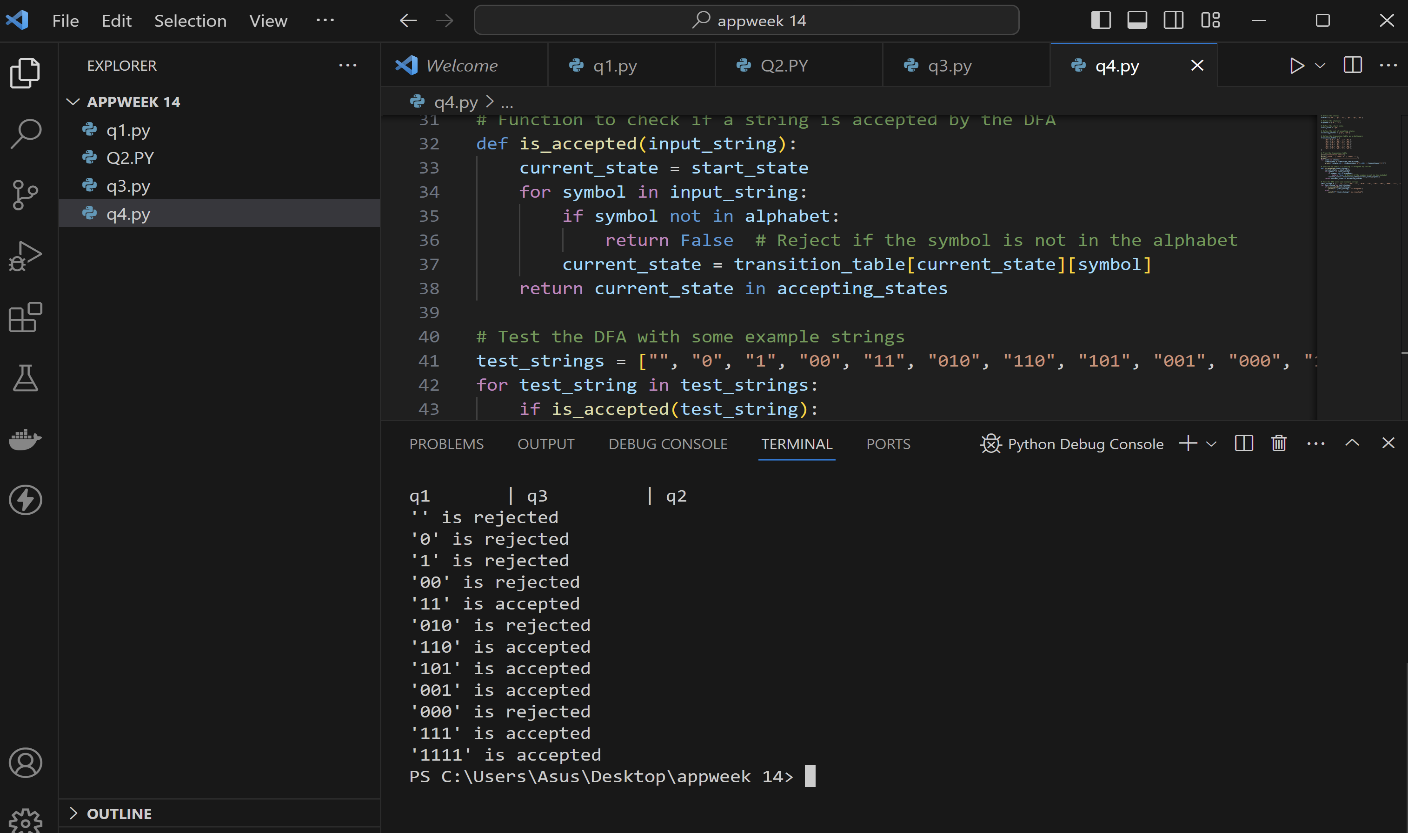


4. Draw a DFA and give the transition table for the language L(M)=a + aa\*b. 5.

Draw a deterministic finite automate which accept 00 and 11 at the end of a

string containing 0, 1

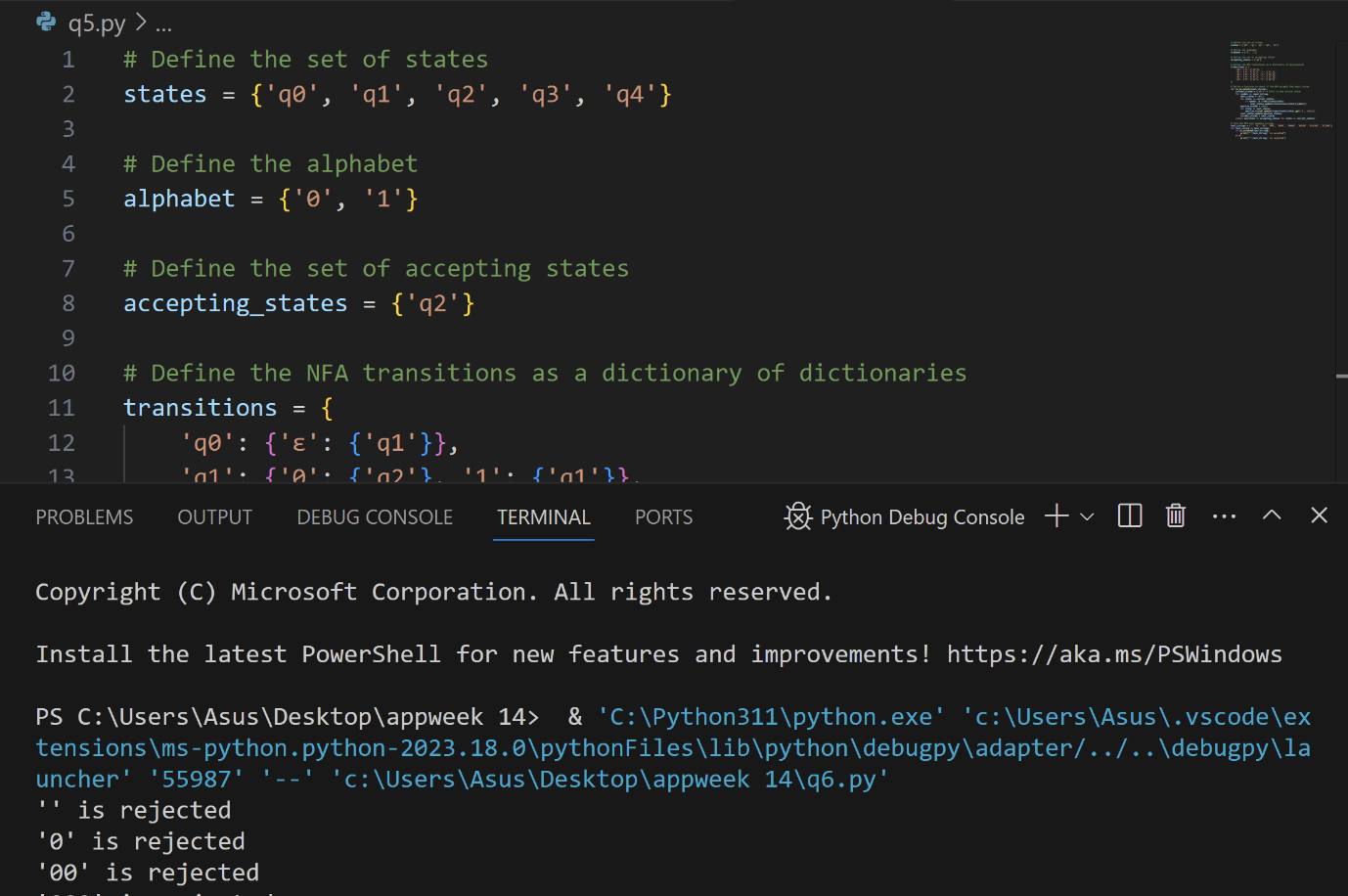




**NFA**

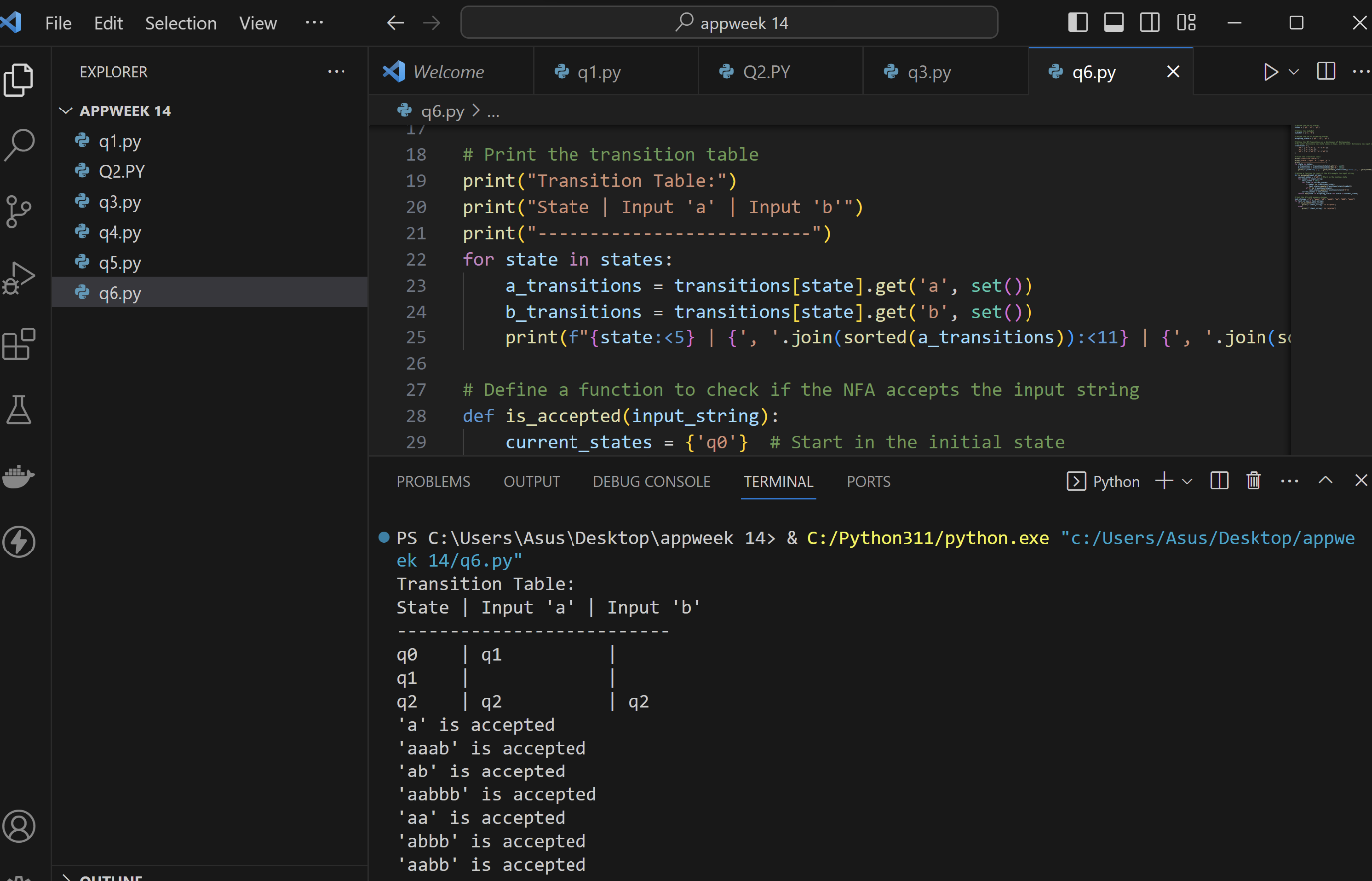
1. NFA recognizes all strings that contain two 0’s separated by a substring whose

length is a multiple of 3.



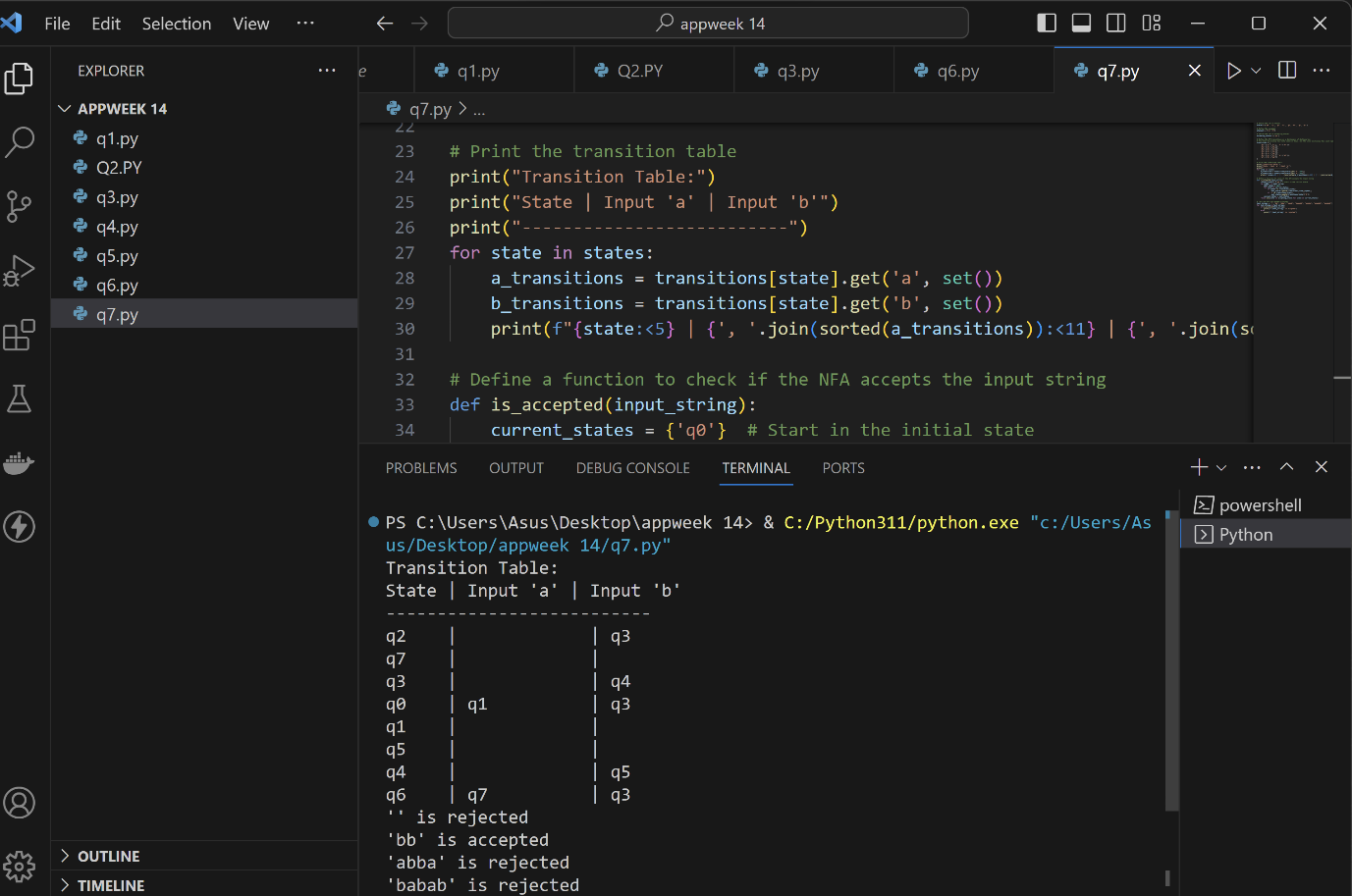
2. Construct a NFA and give the transition table for the following regular expression

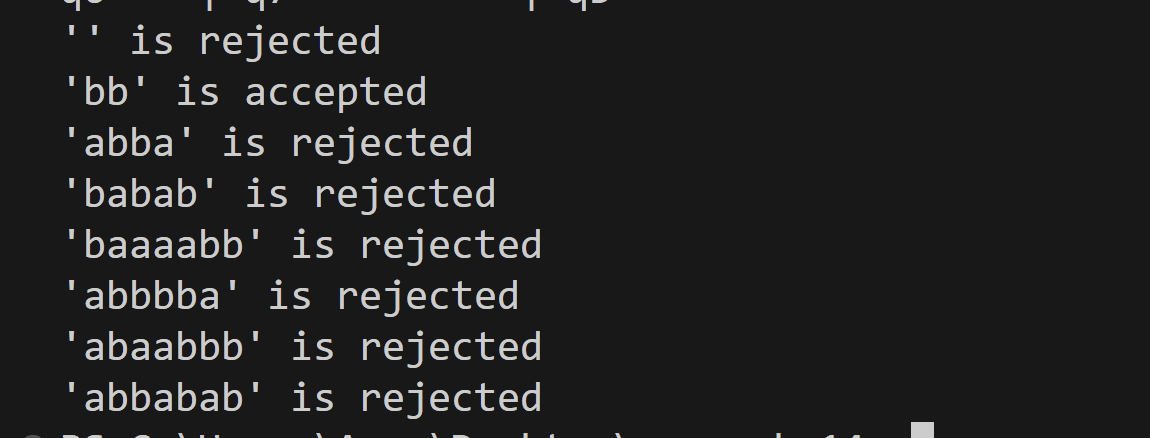
for the given language =a + aa\*b + a\*b.



3. Construct a NFA and give the transition table for the following regular

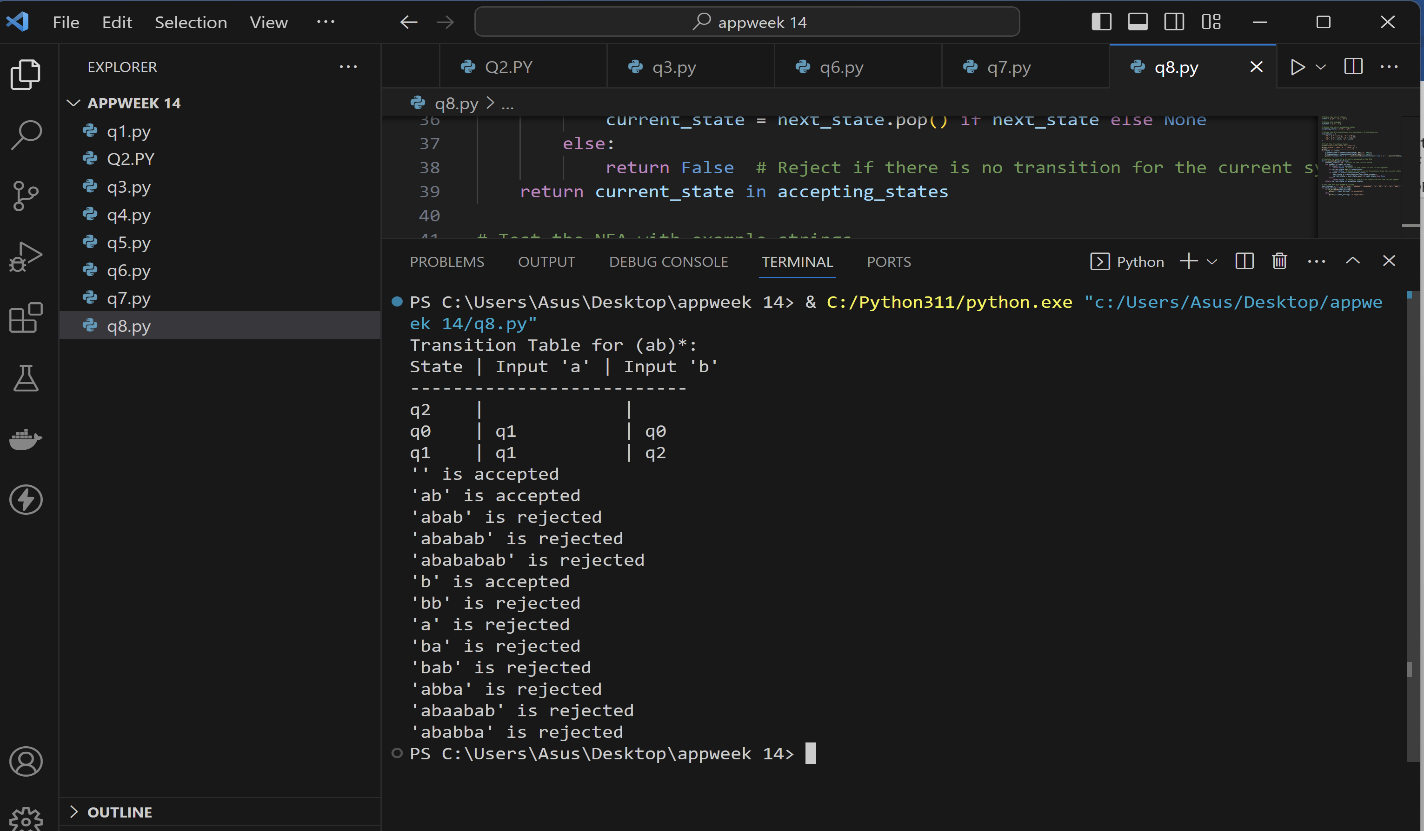
expression for the given language =(a + ba)\*bb(a + ab)\*.





4. Construct a NFAs and give the transition table for {}, {ε}, {(ab)n | n ∈ N}, which

has regular expression (ab)\*.



5. Construct a NFAs and give the transition table for the Language that accepts all

end with 01

