1) Using Python programming language write a DFA machine which accepts the strings to validate any gmail.

Note: Check whether the given mail is valid or not.

String Validation

```
import re

def validate_email(email):
    #DFA States
    states = [
        { "[a-zA-Z0-9._%+-]" : 1},
        { "[a-zA-Z0-9._%+-]" : 1, "@" : 2},
        { "[a-zA-Z0-9.-]" : 3},
        { "[a-zA-Z]" : 4},
        { "[a-zA-Z0-9.-]" : 4}
    ]

    current_state = 0

    for char in email:
        found_transition = False
        for transition, next_state in states[current_state].items():
            if re.match(transition, char):
                current_state = next_state
                found_transition = True
```

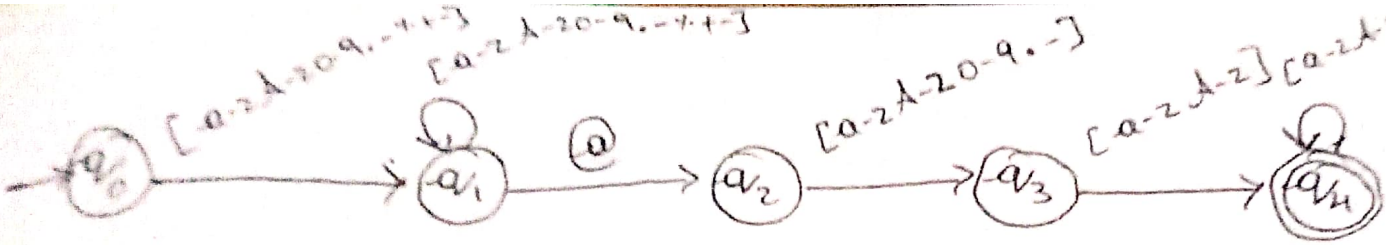
```
    if not found_transition:
        return False
    return current_state in {3,4}

# prompt the user to input an email address
user_email = input("Enter an email address to validate: ")
# validate the user provided email address
if validate_email(user_email):
    print("Valid email address!")
else:
    print("Invalid email address!")
```

L = ik91h149988b@gmail.com

Transition Table

States	1	2	3	4
-> q_0	[a-zA-Z0-9._+-]	ϕ	ϕ	ϕ
q_1	[a-zA-Z0-9._+-]	@	ϕ	ϕ
q_2	ϕ	ϕ	[a-zA-Z0-9.-]	ϕ
q_3	ϕ	ϕ	ϕ	[a-zA-Z]
* q_4	ϕ	ϕ	ϕ	[a-zA-Z0-9.-]



Transition Diagram.

Hence the Language 'L' is accepted.