Write a program using the Regular Exception and create a function that accepts a string and searches it for a valid phone number. Return the phone number if found. A valid phone number may be one of the following: (xxx)-xxx-xxxx xxx-xxx-xxxx

In [ ]:

```
import re

def find_phone_number(text):
    number = re.search(r"(\(\d{3}\)-\d{3}-\d{4}|\d{3}-\d{3}-\d{4})", text)
    if number:
        return number.group()
    return None


phone_number = find_phone_number("Please call me at (123)-456-7890")
print(phone_number)

phone_number = find_phone_number("My number is 987-654-3210")
print(phone_number)

phone_number = find_phone_number("No phone number here")
print(phone_number)

text=input("Enter the phone Number: ")
phone_number = find_phone_number(text)
print(phone_number)
```

```
(123)-456-7890
987-654-3210
None
None
```

Q2. Write a function that employs regular expressions to ensure the password given to the function is strong. A strong password is defined as follows: · at least eight characters long · contains one uppercase character · contains one lowercase character · has at least one digit · has at least one special character

In [ ]:

```
import re

def is_strong_password(password):
    # At least eight characters long
    if len(password) < 8:
        return False

    # Contains one uppercase character
    if not re.search(r'[A-Z]', password):
        return False

    # Contains one lowercase character
    if not re.search(r'[a-z]', password):
        return False

    # Contains at least one digit
    if not re.search(r'\d', password):
        return False

    # Contains at least one special character
    if not re.search(r'[!@#$%^&*()-=_+|;,.<>/?]', password):
        return False

    # All criteria met, password is strong
    return True

# input Christ
password = input("Enter your password: ")
if is_strong_password(password):
    print("Password is strong!")
else:
    print("Password is not strong.")
```

```
Password is strong!
```