```
# ============================================================
# PASSWORD GENERATOR AND VALIDATION SYSTEM
# Mini Project using Python
# ============================================================

import random
import string
import re

# ------------------------------------------------------------
# Function: Generate Password
# ------------------------------------------------------------
def generate_password(length):
    if length < 8:
        print("Password length must be at least 8 characters.")
        return None

    upper = string.ascii_uppercase
    lower = string.ascii_lowercase
    digits = string.digits
    special = "!@#$%^&*()"

    all_chars = upper + lower + digits + special

    # Ensure strong password rules
    password = (
        random.choice(upper) +
        random.choice(lower) +
```

```
        random.choice(digits) +
        random.choice(special)
    )

    for i in range(length - 4):
        password += random.choice(all_chars)

    password = ''.join(random.sample(password, len(password)))
    return password


# ---------------------------------------------------------
# Function: Validate Password
# ---------------------------------------------------------
def validate_password(password):
    errors = []

    if len(password) < 8:
        errors.append("Password length is too short.")

    if not re.search("[A-Z]", password):
        errors.append("Missing uppercase letter.")

    if not re.search("[a-z]", password):
        errors.append("Missing lowercase letter.")

    if not re.search("[0-9]", password):
        errors.append("Missing digit.")
```

```
    if not re.search("[!@#$%^&*()]", password):
        errors.append("Missing special character.")

    if errors:
        return False, errors
    else:
        return True, ["Password is valid and strong."]



# ---------------------------------------------------------
# Function: Password Strength Analysis
# ---------------------------------------------------------
def password_strength(password):
    score = 0

    if len(password) >= 8:
        score += 1
    if re.search("[A-Z]", password):
        score += 1
    if re.search("[a-z]", password):
        score += 1
    if re.search("[0-9]", password):
        score += 1
    if re.search("[!@#$%^&*()]", password):
        score += 1

    if score <= 2:
```

```
        return "WEAK"
    elif score == 3 or score == 4:
        return "MEDIUM"
    else:
        return "STRONG"



# ---------------------------------------------------------
# Function: Testing Module
# ---------------------------------------------------------
def testing_module():
    test_cases = [
        "abc",
        "password",
        "Password1",
        "Password@123",
        "Strong@Pass99"
    ]

    print("\n------ TESTING PASSWORDS ------")
    for pwd in test_cases:
        print("\nPassword:", pwd)
        valid, msg = validate_password(pwd)
        strength = password_strength(pwd)

        print("Strength:", strength)
        if valid:
            print("Validation: PASSED")
```

```
        else:
            print("Validation: FAILED")
            for m in msg:
                print("-", m)



# ---------------------------------------------------------
# Main Menu
# ---------------------------------------------------------
def main():
    while True:
        print("\n==================================")
        print(" PASSWORD GENERATOR & VALIDATOR ")
        print("==================================")
        print("1. Generate Password")
        print("2. Validate Password")
        print("3. Password Strength Check")
        print("4. Testing Module")
        print("5. Exit")

        choice = input("Enter your choice (1-5): ")

        if choice == "1":
            length = int(input("Enter password length: "))
            pwd = generate_password(length)
            if pwd:
                print("Generated Password:", pwd)
                print("Strength:", password_strength(pwd))
```

```
        elif choice == "2":
            pwd = input("Enter password to validate: ")
            valid, messages = validate_password(pwd)
            for m in messages:
                print(m)

        elif choice == "3":
            pwd = input("Enter password: ")
            print("Password Strength:", password_strength(pwd))

        elif choice == "4":
            testing_module()

        elif choice == "5":
            print("Exiting program. Thank you!")
            break

        else:
            print("Invalid choice. Please try again.")


# ------------------------------------------------------------
# Program Execution
# ------------------------------------------------------------
if __name__ == "__main__":
    main()
```