Solution@06-11-23

420. Strong Password Checker

```
import java.util.Scanner;

public class StrongPasswordChecker {
  public static void main(String[] args) {
    Scanner scanner = new Scanner(System.in);

    System.out.println("Please enter a password: ");
    String password = scanner.nextLine();

    String result = checkPasswordStrength(password);

    System.out.println(result);
  }

  public static String checkPasswordStrength(String password) {
    int length = password.length();
    boolean hasLowerCase = false;
    boolean hasUpperCase = false;
    boolean hasDigit = false;

    for (char c : password.toCharArray()) {
      if (Character.isLowerCase(c)) {
        hasLowerCase = true;
      } else if (Character.isUpperCase(c)) {
        hasUpperCase = true;
      } else if (Character.isDigit(c)) {
        hasDigit = true;
      }
```

```
        }

        boolean isLengthValid = length >= 8 && length <= 20;
        boolean hasSpecialCharacters = password.matches(".*[!@#$%^&*()-+=].*");
        boolean hasRepeatedCharacters = password.matches(".*(.)\\1{2,}.*");

        if (isLengthValid && hasLowerCase && hasUpperCase && hasDigit && !hasSpecialCharacters &&
!hasRepeatedCharacters) {
            return "Password is strong!";
        } else {
            StringBuilder message = new StringBuilder("Password is weak due to:\n");
            if (!isLengthValid) {
                message.append("- Length should be between 8 and 20 characters.\n");
            }
            if (!hasLowerCase) {
                message.append("- Should contain at least one lowercase letter.\n");
            }
            if (!hasUpperCase) {
                message.append("- Should contain at least one uppercase letter.\n");
            }
            if (!hasDigit) {
                message.append("- Should contain at least one digit.\n");
            }
            if (hasSpecialCharacters) {
                message.append("- Should not contain special characters like !@#$%^&*()-+=.\n");
            }
            if (hasRepeatedCharacters) {
                message.append("- Should not have repeating characters more than twice in a row.\n");
            }
            return message.toString();
        }
```

```
    }
}
```

Output:-

```
java -cp /tmp/1Gbcvi0hD7 StrongPasswordChecker
Please enter a password:
MyStrongPassword@1234
Password is weak due to:
- Length should be between 8 and 20 characters.
- Should not contain special characters like !@#$%^&*()-+=.
```

507. Perfect Number

```
import java.util.Scanner;

public class PerfectNumber {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);

        System.out.print("Enter a positive integer: ");
        int num = scanner.nextInt();

        if (num <= 0) {
            System.out.println("Please enter a positive integer.");
        } else {
            boolean isPerfect = isPerfectNumber(num);
            if (isPerfect) {
                System.out.println(num + " is a perfect number.");
            } else {
                System.out.println(num + " is not a perfect number.");
            }
        }
    }


    public static boolean isPerfectNumber(int num) {
        int sum = 1; // Initialize with 1 since all numbers are divisible by 1
        for (int i = 2; i <= Math.sqrt(num); i++) {
            if (num % i == 0) {
                sum += i;
                if (i != num / i) {
                    sum += num / i;
                }
            }
        }
        return sum == num;
    }
}
```

Output

```
java -cp /tmp/60wGrAZwyR PerfectNumber
Enter a positive integer: 28
28 is a perfect number.
```