#  Latihan 13.1: Cek Bilangan Prima (Rekursif)

# 💡 Tujuan:

# Cek apakah bilangan n adalah prima atau bukan.

def is\_prime(n, divisor=2):

    if n < 2:

        return False

    if divisor \* divisor > n:

        return True

    if n % divisor == 0:

        return False

    return is\_prime(n, divisor+1)

# Contoh penggunaan

print(is\_prime(7))  # Output: True

print(is\_prime(10)) # Output: False

# Latihan 13.2: Palindrom Rekursif

# 💡 Tujuan:

# Cek apakah sebuah string adalah palindrom.

def is\_palindrome(s):

    s = s.lower().replace(" ", "")  # Hilangkan spasi dan jadikan huruf kecil

    if len(s) <= 1:

        return True

    if s[0] != s[-1]:

        return False

    return is\_palindrome(s[1:-1])

# Contoh penggunaan

print(is\_palindrome("radar"))       # Output: True

print(is\_palindrome("hello"))       # Output: False

print(is\_palindrome("A man a plan a canal Panama"))  # Output: True

# Latihan 13.3: Jumlah Deret Ganjil dari 1 + 3 + 5 + ... + n

# 💡 Tujuan:

# Hitung jumlah deret ganjil dari 1 hingga n (n ganjil).

# ✅ Jawaban:

def jumlah\_deret\_ganjil(n):

    if n == 1:

        return 1

    else:

        return n + jumlah\_deret\_ganjil(n-2)

# Contoh penggunaan

print(jumlah\_deret\_ganjil(7))  # Output: 16 (1+3+5+7=16)

# ✅ Latihan 13.4: Hitung Jumlah Digit Bilangan

# 💡 Tujuan:

# Hitung jumlah digit suatu bilangan.

def jumlah\_digit(n):

    if n == 0:

        return 0

    else:

        return n % 10 + jumlah\_digit(n // 10)

# Contoh penggunaan

print(jumlah\_digit(234))  # Output: 9 (2+3+4)

# Latihan 13.5: Hitung Kombinasi (C(n, r)) Rekursif

# 💡 Rumus Kombinasi:

# ✅ Jawaban (dengan fungsi faktorial rekursif):

def faktorial(n):

    if n == 0 or n == 1:

        return 1

    else:

        return n \* faktorial(n-1)

def kombinasi(n, r):

    return faktorial(n) // (faktorial(r) \* faktorial(n - r))

# Contoh penggunaan

print(kombinasi(5, 2))  # Output: 10

print(kombinasi(10, 4)) # Output: 210

# Contoh Soal Cerita 1: Menjumlahkan Deret Bilangan

# 👧 Soal:

# Dina ingin menghitung jumlah semua bilangan dari 1 sampai n. Buatkan fungsi rekursif untuk membantu Dina.

# 📌 Contoh Kasus:

# Input: n=5 ➜ 1+2+3+4+5=15

def jumlah\_deret(n):

    if n == 1:

        return 1

    else:

        return n + jumlah\_deret(n-1)

print(jumlah\_deret(5))  # Output: 15

#  Contoh Soal Cerita 2: Membalikkan String

# 👧 Soal:

# Rudi ingin membalikkan kata yang dia masukkan secara rekursif. Bantu Rudi membuat programnya!

# 📌 Contoh Kasus:

# Input: "kucing" ➜ Output: "gnicuk"

# ✅ Jawaban:

def balik\_string(s):

    if len(s) == 0:

        return s

    else:

        return balik\_string(s[1:]) + s[0]

print(balik\_string("kucing"))  # Output: gnicuk

#  Contoh Soal Cerita 4: Jumlah Elemen List

# 👧 Soal:

# Lina memiliki daftar angka dan ingin menjumlahkan semua elemennya secara rekursif. Bantu Lina!

# 📌 Contoh Kasus:

# Input: [1, 2, 3, 4] ➜ 10

def jumlah\_list(l):

    if len(l) == 0:

        return 0

    else:

        return l[0] + jumlah\_list(l[1:])

print(jumlah\_list([1, 2, 3, 4]))  # Output: 10

# ✅ Soal Cerita 5: Permutasi Rekursif

# 💡 Soal:

# Buatlah program rekursif untuk menghitung permutasi.

# Rumus: P(n, r) = n! / (n-r)!.

# Contoh: P(10,4) ➜ Output: 5040.

def permutasi(m, n):

    if n == 0:

        return 1

    else:

        return permutasi(m, n-1) \* (m-n+1)

print(permutasi(10, 4))  # Output: 5040

#🔴 Soal Cerita 3: Deret Fibonacci Rekursif

# 👧 Soal:

# Tini ingin mengetahui bilangan ke-n pada deret Fibonacci. Buat fungsi rekursif untuk membantu Tini.

# 📌 Contoh Kasus:

# Input: n=7 ➜ Deret: 1, 1, 2, 3, 5, 8, 13 ➜ Hasil: 13.

def fibo(n):

    if n == 1 or n == 2:

        return 1

    else:

        return fibo(n-1) + fibo(n-2)

print(fibo(7))  # Output: 13

#  Soal Cerita 4: Konversi Basis Rekursif

# 👧 Soal:

# Iwan ingin tahu bagaimana mengubah bilangan desimal ke basis lain (2, 8, atau 16) dengan fungsi rekursif. Bantu Iwan membuat fungsinya!

# 📌 Contoh Kasus:

# Input: angka=9, basis=8 ➜ Hasil: 11.

def toBasis(n, base):

    convertString = "0123456789ABCDEF"

    if n < base:

        return convertString[n]

    else:

        return toBasis(n//base, base) + convertString[n%base]

print(toBasis(9, 8))  # Output: 11