|  |  |
| --- | --- |
|  | **Tugas** |
|  | *Task* |

Mata Kuliah / *Course Name* : **Dasar Pemrograman**

Sesi Pertemuan/ *Session Number* : **IV (Empat)**

Judul Materi / *Title* : **Operator**

Nama Dosen / *Lecturer* : **SHINTA AYUNINGTIAS, S.Kom., M.Kom.**

Jenis Tugas / Type of task : **Case Based ~~/ Project Based~~**

NIM / *Student ID Number* : 20240040032

Nama Mahasiswa/*Student Name* : Ahmad Hasan Maki

Kelas/*Class*  : TI24B

**Study Case Based 4:**

1. Sebutkan dan jelaskan jenis Operator pada Python (*score = 20*)
2. Buatlah sistem untuk memecahkan permasalahan berikut ini: *(score = 20)*
3. Buatlah sistem untuk memecahkan permasalahan berikut ini dalam bentuk biner: (*score = 30*)

diketahui:

x=100

y=10

z=68

1. x & y & z
2. x ^ y << 5 >> 2
3. ~x & ~y | ~z
4. Buatlah sistem untuk memecahkan permasalahan berikut ini dalam bentuk biner: (*score = 30*)

diketahui:

x = 0b1100100

y = 0b110010

z = 0b101



Jawaban

Soal 1

1. **Jenis-Jenis Operator Pada Python**
2. **Aritmatika :**

Untuk Melakukan Operasi Dasar / Umum pada Matematika

1. **Perbandingan :**

UntukMembandingkan nilai dan mengembalikan Boolean(True/False)

1. **Penugasan :**

UntukMenetapkan Nilai pada Variable

1. **Identitas :**

Untuk Memeriksa apakah dua objek merujuk ke memori yang sama/objek yang sama

1. **Logika :**

Untuk Melakukan Operasi Logika Dengan Output Boolean (True/False)

1. **Keanggotaan :**

untuk menguji apakah urutan disajikan dalam suatu objek

(Memeriksa keanggotaan dalam objek iterable)

1. **Bitwise :**

untuk menangani operasi logika bilangan biner dalam bentuk bit.

Selengkap nya Penjelasan Operator Python kami presentasikan di CLI pada menu arimatika di source code

A screenshot of a computer

AI-generated content may be incorrect.

1. Cara Penyelesaian Soal Operasi Aritmatika untuk Soal 2

1. # Step 1: Conversi Soal Agar Sesuai Dengan Simbol Aritmatika

2. # Step 2: Tugas Kan Ke Variable untuk menyimpan Hasil Perhitungan

2. hasil = (200 / (20 - 30)) + (100 \* 21.9\*\*2) / (100 + (10 / 20))

3.

4. # Step 3: Panggil / Print varibable Hasil

5. print(f"Hasil perhitungan: {hasil:.4f}")

6.

1. Cara Penyelesaian Soal Operasi Bitwise

# DEFINISI VARIABEL

x = 100 # 0b01100100

y = 10 # 0b00001010

z = 68 # 0b01000100

# ===================== FUNGSI UTILITAS =====================

def to\_binary(n):

"""Mengonversi angka ke biner dalam format 8-bit"""

return bin(n & 0xFF)[2:].zfill(8) # Mengambil hanya 8-bit terakhir

def interpret\_signed(n):

"""Mengonversi unsigned 8-bit ke signed integer"""

return n if n < 128 else n - 256 # Mengonversi ke signed jika > 127

# ===================== HEADER PROGRAM =====================

print("=" \* 60)

print("=== EVALUASI BERDASARKAN PRESEDENSI OPERATOR PYTHON ===".center(60))

print("=" \* 60)

# Tampilkan nilai awal variabel

print("NILAI AWAL:")

print(f"x = {x} (binary: {to\_binary(x)})")

print(f"y = {y} (binary: {to\_binary(y)})")

print(f"z = {z} (binary: {to\_binary(z)})")

print("-" \* 60)

# ===================== SOAL A =====================

print("SOAL A: Bitwise AND dengan tiga operand (x & y & z)")

print("-" \* 60)

# Evaluasi dari kiri ke kanan

expr\_a = x & y & z

print(f"Hasil: {expr\_a} (binary: {to\_binary(expr\_a)})")

# Tampilkan langkah-langkah evaluasi

print("Langkah evaluasi:")

step1\_result = x & y

print(

f"1. x & y = {step1\_result} ({to\_binary(x)} & {to\_binary(y)} = {to\_binary(step1\_result)})"

)

print(

f"2. (x & y) & z = {expr\_a} ({to\_binary(step1\_result)} & {to\_binary(z)} = {to\_binary(expr\_a)})"

)

print("-" \* 60)

# ===================== SOAL B1 =====================

print("SOAL B1: Kombinasi operator bitwise tanpa kurung eksplisit")

print("Ekspresi: x ^ y << 5 >> 2")

print("-" \* 60)

# Presedensi: shift, XOR, shift

expr\_b\_no\_parens = x ^ y << 5 >> 2

print(f"Hasil: {expr\_b\_no\_parens} (binary: {to\_binary(expr\_b\_no\_parens)})")

# Tampilkan langkah-langkah evaluasi berdasarkan presedensi

print("Langkah evaluasi berdasarkan presedensi Python:")

step1\_b1 = y << 5

print(f"1. y << 5 = {step1\_b1} ({to\_binary(y)} << 5 = {to\_binary(step1\_b1)})")

step2\_b1 = x ^ step1\_b1

print(

f"2. x ^ (y << 5) = {step2\_b1} ({to\_binary(x)} ^ {to\_binary(step1\_b1)} = {to\_binary(step2\_b1)})"

)

print(

f"3. (x ^ (y << 5)) >> 2 = {expr\_b\_no\_parens} ({to\_binary(step2\_b1)} >> 2 = {to\_binary(expr\_b\_no\_parens)})"

)

print("-" \* 60)

# ===================== SOAL B2 =====================

print("SOAL B2: Kombinasi operator bitwise dengan kurung eksplisit")

print("Ekspresi: x ^ (y << 5) >> 2")

print("-" \* 60)

# Langkah-langkah eksplisit

y\_shift = y << 5 # Geser y ke kiri 5 bit

xor\_result = x ^ y\_shift # XOR dengan x

expr\_b\_parens = xor\_result >> 2 # Geser hasil XOR ke kanan 2 bit

print(f"Hasil: {expr\_b\_parens} (binary: {to\_binary(expr\_b\_parens)})")

# Tampilkan langkah-langkah evaluasi dengan kurung eksplisit

print("Langkah evaluasi dengan kurung eksplisit:")

print(f"1. y << 5 = {y\_shift} ({to\_binary(y)} << 5 = {to\_binary(y\_shift)})")

print(

f"2. x ^ (y << 5) = {xor\_result} ({to\_binary(x)} ^ {to\_binary(y\_shift)} = {to\_binary(xor\_result)})"

)

print(

f"3. (x ^ (y << 5)) >> 2 = {expr\_b\_parens} ({to\_binary(xor\_result)} >> 2 = {to\_binary(expr\_b\_parens)})"

)

print("-" \* 60)

# ===================== SOAL C =====================

print("SOAL C: Kombinasi bitwise NOT, AND, dan OR")

print("Ekspresi: ~x & ~y | ~z")

print("-" \* 60)

# Negasi bitwise (dalam batas 8-bit)

not\_x = ~x & 0xFF

not\_y = ~y & 0xFF

not\_z = ~z & 0xFF

# Evaluasi berdasarkan presedensi

expr\_c = (not\_x & not\_y) | not\_z

# Interpretasi hasil

result\_signed = interpret\_signed(expr\_c)

print(f"Hasil signed: {result\_signed} (binary: {to\_binary(expr\_c)})")

print(f"Hasil unsigned: {expr\_c} (binary: {to\_binary(expr\_c)})")

# Tampilkan langkah-langkah evaluasi

print("Langkah evaluasi:")

print(f"1. ~x = {not\_x} (binary: {to\_binary(not\_x)})")

print(f"2. ~y = {not\_y} (binary: {to\_binary(not\_y)})")

print(f"3. ~z = {not\_z} (binary: {to\_binary(not\_z)})")

step4\_c = not\_x & not\_y

print(

f"4. ~x & ~y = {step4\_c} ({to\_binary(not\_x)} & {to\_binary(not\_y)} = {to\_binary(step4\_c)})"

)

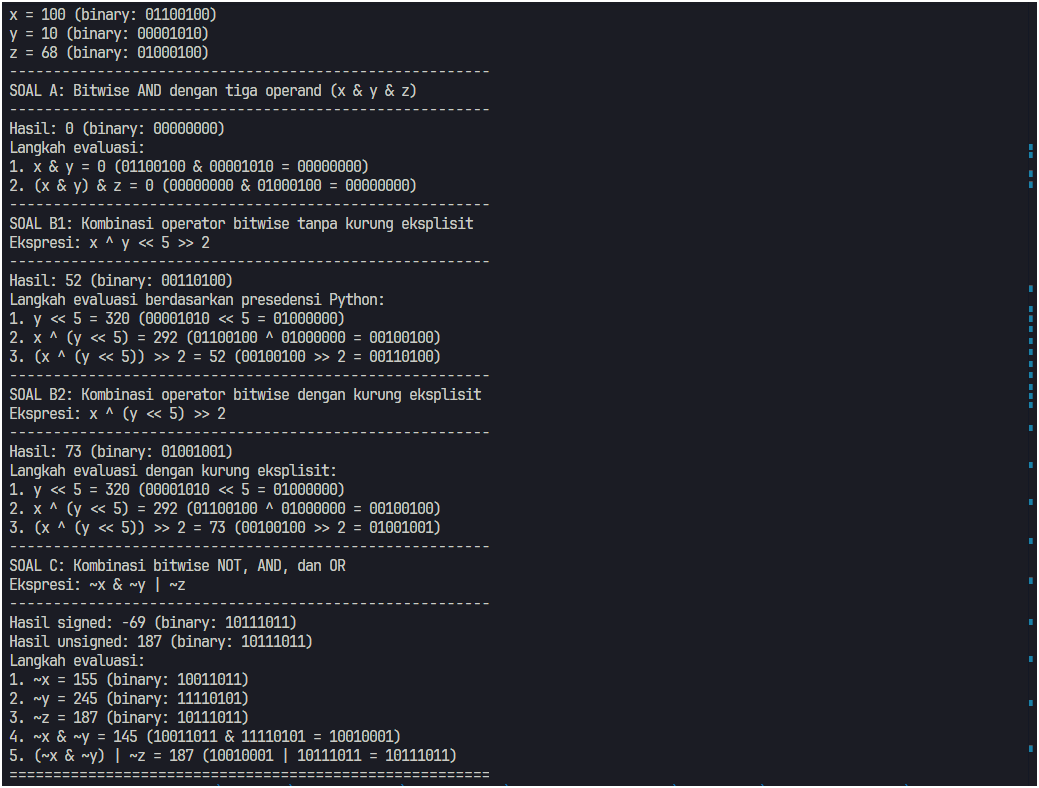
print(

f"5. (~x & ~y) | ~z = {expr\_c} ({to\_binary(step4\_c)} | {to\_binary(not\_z)} = {to\_binary(expr\_c)})"

)

print("=" \* 60)

Output hasil nya adalah :



1. Penyelesaian Soal Nomor 4

# Mendefinisikan variabel

x = 0b1100100 # 100 dalam desimal

y = 0b110010 # 50 dalam desimal

z = 0b101 # 5 dalam desimal

print("=== NILAI AWAL VARIABEL ===")

print(f"x = {x} (biner: {bin(x)})")

print(f"y = {y} (biner: {bin(y)})")

print(f"z = {z} (biner: {bin(z)})")

print("-" \* 50)

# Fungsi untuk mengonversi desimal ke biner

def to\_binary(n):

if n == 0:

return "0b0"

is\_negative = n < 0

if is\_negative:

n = -n # Ubah ke positif untuk konversi

binary = bin(n)[2:] # Hilangkan prefiks '0b'

return "-0b" + binary if is\_negative else "0b" + binary

# ========= EKSPRESI A: (x - y) / z =========

print("\n=== EKSPRESI A: (x - y) / z ===")

print("Langkah-langkah perhitungan:")

# Langkah 1: Kurangkan y dari x

a\_numerator = x - y

print(f"1. x - y = {x} - {y} = {a\_numerator}")

print(f" Dalam biner: {bin(x)} - {bin(y)} = {bin(a\_numerator)}")

# Langkah 2: Bagi hasil dengan z

a\_result = a\_numerator / z

print(f"2. (x - y) / z = {a\_numerator} / {z} = {a\_result}")

# Tampilkan hasil pembulatan dan konversi ke biner

a\_rounded = round(a\_result)

a\_int = int(a\_result)

print("\nHasil Ekspresi a:")

print(f"Desimal (tidak dibulatkan): {a\_result}")

print(f"Desimal (dibulatkan ke integer terdekat): {a\_rounded}")

print(f"Desimal (dikonversi ke integer): {a\_int}")

print(f"Biner (dari konversi ke integer): {to\_binary(a\_int)}")

print(

f"Apakah hasil adalah bilangan bulat? {'Ya' if a\_result.is\_integer() else 'Tidak'}"

)

# ========= EKSPRESI B: ((x \* z) + y) / (x - z + z) =========

print("\n=== EKSPRESI B: ((x \* z) + y) / (x - z + z) ===")

print("Langkah-langkah perhitungan:")

# Langkah 1: Kalikan x dengan z

b\_step1 = x \* z

print(f"1. x \* z = {x} \* {z} = {b\_step1}")

print(f" Dalam biner: {bin(x)} \* {bin(z)} = {bin(b\_step1)}")

# Langkah 2: Tambahkan y ke hasil

b\_numerator = b\_step1 + y

print(f"2. (x \* z) + y = {b\_step1} + {y} = {b\_numerator}")

print(f" Dalam biner: {bin(b\_step1)} + {bin(y)} = {bin(b\_numerator)}")

# Langkah 3: Hitung penyebut (x - z + z)

b\_step3\_sub = x - z

print(f"3a. x - z = {x} - {z} = {b\_step3\_sub}")

print(f" Dalam biner: {bin(x)} - {bin(z)} = {bin(b\_step3\_sub)}")

b\_denominator = b\_step3\_sub + z

print(f"3b. (x - z) + z = {b\_step3\_sub} + {z} = {b\_denominator}")

print(f" Dalam biner: {bin(b\_step3\_sub)} + {bin(z)} = {bin(b\_denominator)}")

print(f" Catatan: x - z + z disederhanakan menjadi x (yaitu {x})")

# Langkah 4: Bagi

b\_result = b\_numerator / b\_denominator

print(f"4. ((x \* z) + y) / (x - z + z) = {b\_numerator} / {b\_denominator} = {b\_result}")

# Tampilkan hasil pembulatan dan konversi ke biner

b\_rounded = round(b\_result)

b\_int = int(b\_result)

print("\nHasil Ekspresi b:")

print(f"Desimal (tidak dibulatkan): {b\_result}")

print(f"Desimal (dibulatkan ke integer terdekat): {b\_rounded}")

print(f"Desimal (dikonversi ke integer): {b\_int}")

print(f"Biner (dari konversi ke integer): {to\_binary(b\_int)}")

print(

f"Apakah hasil adalah bilangan bulat? {'Ya' if b\_result.is\_integer() else 'Tidak'}"

)

# Perhatikan bahwa (x - z + z) = x, sehingga ekspresi b menjadi ((x \* z) + y) / x

print("\n=== CATATAN PENTING ===")

print("Dalam Ekspresi B: ((x \* z) + y) / (x - z + z)")

print(

"Perhatikan bahwa (x - z + z) = x, sehingga ekspresi dapat disederhanakan menjadi:"

)

print("((x \* z) + y) / x")

simplified\_result = ((x \* z) + y) / x

print(f"Hasil: ((x \* z) + y) / x = {simplified\_result}")

print(f"Yang sama dengan hasil sebelumnya: {b\_result}")

