

#### PSEUCODE KPK:

1. Cari nilai KPK dari 3 dan 4

```
def gcd(a, b):
```

```
while b != 0:
```

```
    remainder = a % b  
    a = b
```

```
    b = remainder  
return a
```

```
def lcm(a, b):
```

```
    return (a * b) // gcd(a, b)
```

```
def main():  
    number1 = 3
```

```
    number2 = 4
```

```
    kpk = lcm(number1, number2)
```

```
    print(f"KPK dari {number1} dan {number2} adalah {kpk}")  
main()
```

#### Algoritma dari KPK:

1. Mulai
2. Tentukan Bilangan bulat dari 'a' dan "b"
3. Definisikan sebuah fungsi bernama 'find\_lcm' dengan parameter 'a' dan 'b':
  - Periksa apakah 'a' lebih besar dari 'b'. Jika iya, berikan nilai 'greater' sama dengan 'a'; jika tidak, berikan nilai 'greater' sama dengan 'b'.
  - Masuk ke dalam loop while:
  - Dalam loop while, periksa apakah 'greater' habis dibagi oleh 'a' dan 'b' (artinya 'greater' adalah kelipatan dari keduanya).
  - Jika kondisi terpenuhi, maka 'greater' adalah KPK dari 'a' dan 'b'. Keluar dari loop dengan mengatur nilai 'lcm' sama dengan 'greater', lalu hentikan loop dengan pernyataan 'break'.
  - Jika kondisi belum terpenuhi, tambahkan 1 ke 'greater' untuk melanjutkan pencarian kelipatan berikutnya.
4. Setelah keluar dari loop, kembalikan nilai 'lcm' sebagai hasil dari fungsi 'find\_lcm'.
5. Tetapkan dua bilangan 'a' dan 'b' dengan nilai masing-masing (contohnya a = 3 dan b = 4).
6. Panggil fungsi 'find\_lcm' dengan argumen 'a' dan 'b', dan simpan hasilnya dalam variabel 'kpk'.
7. Cetak pesan yang berisi Nilai KPK "a" Dan "b" menggunakan pernyataan dari 'print'
8. Finish/Selesai

Pseuocode Menukar Posisi X dan Y dengan Kasus:

ada 2 buah: manggis dan pisang, manggis di piring 1, pisang di piring 2, piring 3 kosong

p1 = buah manggis

p2 = buah pisang

p3 = null/kosong

define swap(p1,p2)

p3 = p1

p1 = p2

p2 = p3

Algoritma:

1. Pindahkan buah manggis ke piring no 3 dan piring no 1 menjadi null
2. Pindahkan buah pisang ke piring no 1 dan piring no 2 menjadi null
3. Pindahkan buah manggis ke piring no 2 dan piring no 3 menjadi null

Diketahui sebuah segitiga memiliki ukuran sebagai berikut:

Alas 25

Tinggi 30

Hitung luas dari segitiga tersebut.

Pseudocodenya:

Alas:25

Tinggi:30

Luas=  $\frac{1}{2}$  x alas x tinggi

Luas= $\frac{1}{2}$  x 25x30

Print (Luas segitiga)

Algoritma:

1. Deklarasikan luas, alas, dan tinggi
2. Masukkan Alas=25, Tinggi=30
3. Menghitung luas= $\frac{1}{2}$  x alas x tinggi
4. Menghitung Luas= $\frac{1}{2}$  x 25 x 30
5. Menampilkan Luas Segitiga

Tentukan algoritma dan pseudocode luas jajar genjang

(panjang = 5, tinggi = 3)

Pseudocode:

panjang = 5

tinggi = 3

luas = panjang \* tinggi

print("Luas jajar genjang:", luas)

Algoritma:

1. Mulai
2. Masukkan nilai Panjang dari jajar genjang
3. Masukkan nilai Tinggi jajar genjang
4. Hitung luas dengan rumus jajar genjang
5. Nilai dari luas jajar genjang muncul
6. Selesai/Finish

Tentukan algoritma dan pseudocode volume tabung (jari-jari = 3, tinggi = 5)

Pseudocode:

```
jari_jari = 3
```

```
tinggi = 5
```

```
pi = 3,14
```

```
volume = pi * jari_jari ** 2 * tinggi
```

```
print("Volume tabung:", volume)
```

Algoritma:

1. Mulai
2. Masukkan nilai jari jari
3. Masukkan nilai tinggi
4. Masukkan nilai pi=3.14
5. Hitung Volume tabung dengan rumus  $3.14 \times 3 \times 5$
6. Nilai dari volume tabung muncul
7. Selesai

Tentukan algoritma dan pseudocode volume kerucut

(diameter = 5, tinggi = 4)

Pseudocode:

Diameter=5

Tinggi=4

Jari jari= diameter/2

Pi:3.14

Volume =  $(1/3) * \pi * \text{jari\_jari}^2 * \text{tinggi}$

Print("Volume kerucut:", volume)

Algoritma:

1. Mulai
2. Masukkan nilai pi,diameter,jari jari,tinggi
3. Hitung volume kerucut dengan rumus:  $\text{volume} = ((1/3) * \pi * \text{jari\_jari}^2 * \text{tinggi})$
4. Nilai hasil volume kerucut muncul
5. Finish/Selesai