

- Skala peta = jarak pada peta : jarak sebenarnya  
 $Sp = 10 \text{ cm} : 220 \text{ km} (0,002200000 \text{ cm}) = 4545,45454545455 \text{ cm}$

- Konsep 1

The screenshot displays a Python program for calculating map scale. It includes a flowchart, source code, and terminal output.

**Flowchart:**

```

graph TD
    Main([Main]) --> RealSpJpJs[Real sp, jp, js]
    RealSpJpJs --> OutputJp[Output "jarak peta=10cm"]
    OutputJp --> Jp10[jp = 10]
    Jp10 --> OutputJs[Output "jarak sebenarnya=220km"]
    OutputJs --> OutputJsCm[Output "jarak sebenarnya dalam bentuk satuan cm= 0,002200000"]
    OutputJsCm --> Js00022[js = 0.002200000]
    Js00022 --> SpJpJs[sp = jp/js]
    SpJpJs --> OutputSp[Output "skala pada peta tersebut adalah " + str(sp)]
  
```

**Source Code (konsep 1 skala peta.py):**

```

0 print("jarak peta=10cm")
1 jp = 10
2 print("jarak sebenarnya=220km")
3 print("jarak sebenarnya dalam bentuk satuan cm= 0,002200000")
4 js = 0.002200000
5 sp = jp / js
6 print("skala pada peta tersebut adalah " + str(sp), end="")
7 print("cm")
  
```

**Terminal Output:**

```

PS C:\Users\ASUS> & C:\Users\ASUS\AppData\Local\Programs\Python\Python310\python.exe "c:\Users\ASUS\Documents\praktikum 4\no 1 skala peta\konsep 1\konsep 1 skala peta.py"
jarak peta=10cm
jarak sebenarnya=220km
jarak sebenarnya dalam bentuk satuan cm= 0,002200000
skala pada peta tersebut adalah 4545.45454545455cm
  
```

- Konsep 2

The screenshot displays a Python program for calculating map scale with user input. It includes a flowchart, source code, and terminal output.

**Flowchart:**

```

graph TD
    Main([Main]) --> RealSpJpJs[Real sp, jp, js]
    RealSpJpJs --> OutputJp[Output "masukkan jarak pada peta"]
    OutputJp --> InputJp[Input jp]
    InputJp --> OutputJs[Output "masukkan jarak sebenarnya"]
    OutputJs --> InputJs1[Input js]
    InputJs1 --> OutputJsCm[Output "masukkan dalam satuan cm"]
    OutputJsCm --> InputJs2[Input js]
    InputJs2 --> SpJpJs[sp = jp/js]
  
```

**Source Code (konsep 2 skala peta.py):**

```

0 print("masukkan jarak pada peta")
1 jp = float(input())
2 print("masukkan jarak sebenarnya")
3 js = float(input())
4 print("masukkan dalam satuan cm")
5 js = float(input())
6 sp = jp / js
7 print("skala pada peta tersebut adalah " + str(sp), end="")
8 print("cm")
  
```

**Terminal Output:**

```

PS C:\Users\ASUS> & C:\Users\ASUS\AppData\Local\Programs\Python\Python310\python.exe "c:\Users\ASUS\Documents\praktikum 4\no 1 skala peta\konsep 2\konsep 2 skala peta.py"
masukkan jarak pada peta
10
masukkan jarak sebenarnya
220
masukkan dalam satuan cm
0.002200000
skala pada peta tersebut adalah 4545.45454545455cm
  
```

2. Jarak sebenarnya = jarak peta \* skala  
 $J_s = 10 \text{ cm} * 1:5.000.000 = 100.000.000 \text{ cm (1.000 km)}$

- Konsep 1

The screenshot displays a Python IDE with three main components:

- Flowchart (Left):** A flowchart titled 'Main' showing the logic of the program. It starts with 'Real s.jp.js', followed by 'Output "jarak pada peta=20 cm"', then 'jp = 20', 'Output "dengan skala= 1:5000000"', 's = 5000000', 'js = jp \* s', 'Output "jarak sebenarnya adalah &js ..."', and finally 'Output "cm"'. The file was saved at 12:46 AM.
- Source Code (Middle):** A Python script with the following code:

```
0 print("jarak pada peta=20 cm")
1 jp = 20
2 print("dengan skala= 1:5000000")
3 s = 5000000
4 js = jp * s
5 print("jarak sebenarnya adalah " + str(js), end='
6 print("cm")
```
- Terminal (Right):** The output of the program execution in a Windows PowerShell window:

```
PS C:\Users\ASUS> & C:\Users\ASUS\AppData\Local\Programs\Python\Python310\python.exe "c:/Users/ASUS/Documents/praktikum 4/no 2 jarak sesungguhnya/konsep 1/konsep 1 jarak sesungguhnya.py"
jarak pada peta=20 cm
dengan skala= 1:5000000
jarak sebenarnya adalah 100000000cm
PS C:\Users\ASUS>
```

- Konsep 2

The screenshot displays a Python IDE with three main components:

- Flowchart (Left):** A flowchart titled 'Main' showing the logic of the program. It starts with 'Real s.jp.js', followed by 'Output "masukkan jarak pada peta"', 'Input jp', 'Output "masukkan skala"', 'Input s', 'js = jp \* s', 'Output "jarak sebenarnya adalah &js ..."', 'Output "cm"', and finally 'End'. The file was saved at 12:04 AM.
- Source Code (Middle):** A Python script with the following code:

```
0 print("masukkan jarak pada peta")
1 jp = float(input())
2 print("masukkan skala")
3 s = float(input())
4 js = jp * s
5 print("jarak sebenarnya adalah " + str(js), end='
6 print("cm")
```
- Terminal (Right):** The output of the program execution in a Windows PowerShell window:

```
PS C:\Users\ASUS> & C:\Users\ASUS\AppData\Local\Programs\Python\Python310\python.exe "c:/Users/ASUS/Documents/praktikum 4/no 2 jarak sesungguhnya/konsep 2/konsep 2 jarak sesungguhnya.py"
masukkan jarak pada peta
20
masukkan skala
5000000
jarak sebenarnya adalah 100000000.0cm
PS C:\Users\ASUS>
```

3. Jarak = perwakilan pada peta \* ruas

Jarak = 1 km \* 4 ruas = 4 km

- Konsep 1

The screenshot displays a Python IDE with three main components: a flowchart, a source code editor, and a terminal window.

**Flowchart (konsep 1 jarak - Flowgorithm):**

```
graph TD
    Main([Main]) --> Real[Real jarak, ruas]
    Real --> Output1[/Output "perwakilan ruas= 1 km"/]
    Output1 --> Ruas1[ruas = 1]
    Ruas1 --> Output2[/Output "ruas pada peta= 4 ruas"/]
    Output2 --> Ruas4[ruas = 4]
    Ruas4 --> Jarak[jarak = ruas * 1]
    Jarak --> Output3[/Output "jarak antara kedua kecamatan tersebut adalah "& jarak .../]
    Output3 --> Output4[/Output "km"/]
```

**Source Code (konsep 1 jarak.py):**

```
0 print("perwakilan ruas= 1 km")
1 ruas = 1
2 print("ruas pada peta= 4 ruas")
3 ruas = 4
4 jarak = ruas * 1
5 print("jarak antara kedua kecamatan tersebut adalah ")
6 print("km")
```

**Terminal Output:**

```
PS C:\Users\ASUS> & C:\Users\ASUS\AppData\Local\Programs\Python\Python310\python.exe "c:/Users/ASUS/Documents/praktikum 4/no 3 jarak/konsep 1 jarak.py"
perwakilan ruas= 1 km
ruas pada peta= 4 ruas
jarak antara kedua kecamatan tersebut adalah 4km
PS C:\Users\ASUS>
```

- Konsep 2

The screenshot displays a Python IDE with three main components: a flowchart, a source code editor, and a terminal window.

**Flowchart (konsep 2 jarak - Flowgorithm):**

```
graph TD
    Main([Main]) --> Real[Real jarak, ruas]
    Real --> Output1[/Output "masukkan perwakilan ruas"/]
    Output1 --> Input1[/Input ruas/]
    Input1 --> Output2[/Output "masukkan ruas pada peta"/]
    Output2 --> Input2[/Input ruas/]
    Input2 --> Jarak[jarak = ruas * 1]
    Jarak --> Output3[/Output "jarak antara kedua kecamatan tersebut adalah "& jarak .../]
    Output3 --> Output4[/Output "km"/]
```

**Source Code (konsep 2 jarak.py):**

```
0 print("masukkan perwakilan ruas")
1 ruas = float(input())
2 print("masukkan ruas pada peta")
3 ruas = float(input())
4 jarak = ruas * 1
5 print("jarak antara kedua kecamatan tersebut adalah ")
6 print("km")
```

**Terminal Output:**

```
PS C:\Users\ASUS> & C:\Users\ASUS\AppData\Local\Programs\Python\Python310\python.exe "c:/Users/ASUS/Documents/praktikum 4/no 3 jarak sesungguhnya/konsep 2 jarak.py"
masukkan perwakilan ruas
1
masukkan ruas pada peta
4
jarak antara kedua kecamatan tersebut adalah 4.0km
PS C:\Users\ASUS>
```