

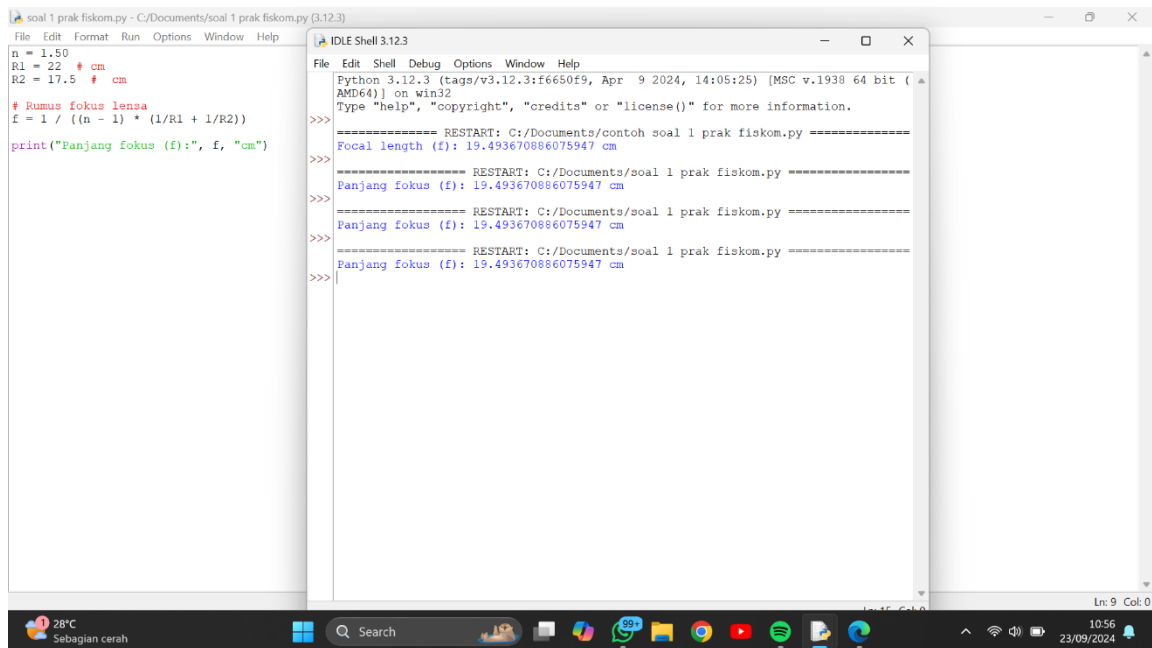
# PRAKTIKUM FISIKA KOMPUTASI

## Penyelesaian Soal Fisika Menggunakan Pemrograman

Eneng Yulia Pebryanti

NIM 1227030013

1. Penjelasan algoritma pemrograman yang digunakan untuk menyelesaikan soal
  - a. Hitung jarak fokus lensa (f)



```
soal 1 prak fisikom.py - C:/Documents/soal 1 prak fisikom.py (3.12.3)
File Edit Format Run Options Window Help
n = 1.50
R1 = 22 # cm
R2 = 17.5 # cm
# Rumus fokus lensa
f = 1 / ((n - 1) * (1/R1 + 1/R2))
print("Panjang fokus (f):", f, "cm")

IDLE Shell 3.12.3
File Edit Shell Debug Options Window Help
Python 3.12.3 (tags/v3.12.3:f6650f9, Apr 9 2024, 14:05:25) [MSC v.1938 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:/Documents/contoh soal 1 prak fisikom.py =====
Focal length (f): 19.493670886075947 cm
>>>
===== RESTART: C:/Documents/soal 1 prak fisikom.py =====
Panjang fokus (f): 19.493670886075947 cm
>>>
===== RESTART: C:/Documents/soal 1 prak fisikom.py =====
Panjang fokus (f): 19.493670886075947 cm
>>>
===== RESTART: C:/Documents/soal 1 prak fisikom.py =====
Panjang fokus (f): 19.493670886075947 cm
>>>
```

Program dimulai dengan mendefinisikan soal yang diketahui pada modul

- n yang menyimpan nilai indeks bias lensa (1.50)
- R1 yang menyimpan jari-jari kelengkungan permukaan pertama lensa (22 cm)
- R2 yang menyimpan jari-jari kelengkungan permukaan kedua lensa (17.5 cm)

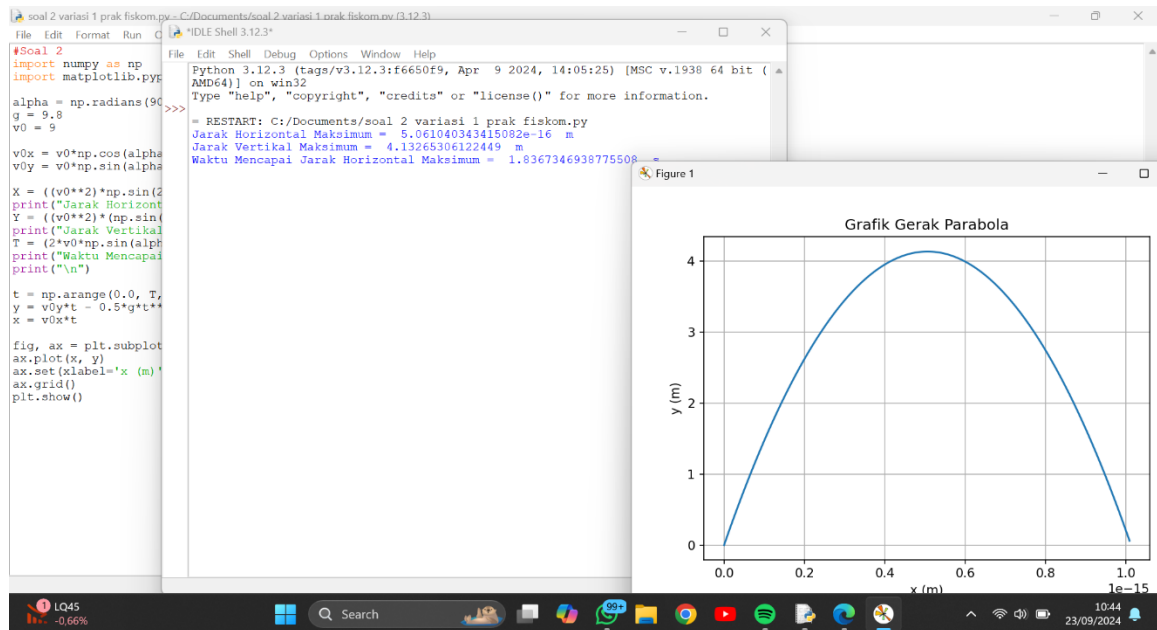
Program kemudian melakukan perhitungan jarak fokus (f) menggunakan rumus lensa

$$\frac{1}{f} = (n - 1) \left[ \frac{1}{R_1} + \frac{1}{R_2} \right]$$

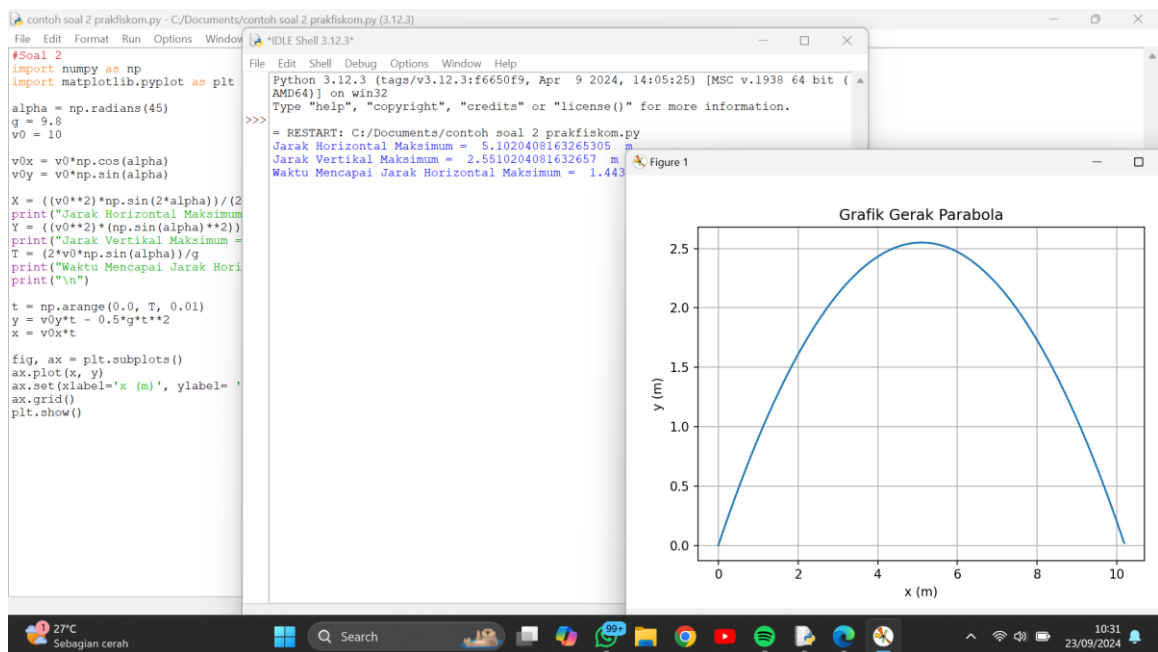
Setelah program menghitung panjang fokus (f), hasil tersebut ditampilkan ke layar menggunakan perintah `print("Panjang fokus (f):", f, "cm")`. Program dijalankan dengan menekan tombol **Run** lalu *software* akan mulai melakukan perhitungan. Setelah program selesai menghitung, hasilnya ditampilkan di layar, yaitu panjang fokus lensa (19.49 cm).

## b. Menghitung jarak horizontal dan vertikal pada gerak parabola

### variasi 1



### variasi 2



Pertama mengimpor library menggunakan dua library, yaitu numpy dan matplotlib. numpy digunakan untuk melakukan perhitungan, sementara matplotlib digunakan untuk memunculkan grafik. Selanjutnya menentukan

### Variasi 1

- sudut (alpha) = 90
- percepatan gravitasi bumi (g) = 9.8
- kecepatan awal (v0) = 9

### Variasi 2

- sudut (alpha) = 45
- percepatan gravitasi bumi (g) = 9.8
- kecepatan awal (v0) = 10
- v0x: kecepatan horizontal, menggunakan rumus  $v_0 * \cos(\alpha)$ .
- v0y: kecepatan vertikal, menggunakan  $v_0 * \sin(\alpha)$ . karena gerak parabola merupakan gerakan ke samping dan ke atas.
- Lalu memasukan rumus berikut untuk menghitung Jarak Horizontal dan Vertikal Maksimum dan waktu

```
X = ((v0**2)*np.sin(2*alpha))/(2*g)
print("Jarak Horizontal Maksimum = ",X, " m")
Y = ((v0**2)*(np.sin(alpha)**2))/(2*g)
print("Jarak Vertikal Maksimum = ",Y, " m")
T = (2*v0*np.sin(alpha))/g
print("Waktu Mencapai Jarak Horizontal Maksimum = ",T, " s")
print("\n")
```

Tinggi Maksimum

$$Y = \frac{v_0^2 \sin^2 \alpha}{2g}$$

Jarak Maksimum

$$X = \frac{v_0^2 \sin 2\alpha}{2g}$$

Waktu Mencapai Jarak Maksimum

$$T = \frac{2v_0 \sin \alpha}{g}$$

- Kode print () menampilkan hasil perhitungan jarak horizontal, vertikal, dan waktu tempuh untuk mencapai jarak horizontal maksimum.
- Kemudian bagian kode program ini untuk menghitung pergerakan benda pada range waktu 0 hingga 0.01 detik. Y dan x adalah rumus untuk menghitung posisi

```
t = np.arange(0.0, T, 0.01)
y = v0y*t - 0.5*g*t**2
x = v0x*t
```

- Kode program terakhir dibuat untuk membuat dan menampilkan grafik

```
fig, ax = plt.subplots()
ax.plot(x, y)
ax.set(xlabel='x (m)', ylabel='y (m)', title='Grafik Gerak Parabola')
ax.grid()
plt.show()
```

- Lalu menjalankan running untuk menampilkan hasil nilai nilai jarak horizontal maksimum, jarak vertikal maksimum, dan waktu tempuh maksimum. Dan menunjukkan grafik lintasan gerak parabola.