DOKUMENTASI PROGRAM PYTHON

”CENTER OF MASS + VELOCITY”

A logo with text on it

Description automatically generated

Disusun oleh:

Yohanes Dimas Pratama

A11.2021.13254

PROGRAM STUDI TEKNIK INFORMATIKA

FAKULTAS ILMU KOMPUTER

UNIVERSITAS DIAN NUSWANTORO

DAFTAR ISI

[BAB I – DASAR TEORI 3](#_Toc171883676)

[1.1 Pusat Massa 3](#_Toc171883677)

[1.2 Kecepatan Pusat Massa 3](#_Toc171883678)

[BAB II - DEMO APLIKASI 4](#_Toc171883679)

[2.1 Link Video Demo Aplikasi 4](#_Toc171883680)

[2.2 Link Google Colab 4](#_Toc171883681)

[2.3 Link GitHub 4](#_Toc171883682)

[2.4 Code Program Menggunakan Google Collab 4](#_Toc171883683)

[2.5 Hasil Menjalankan Menu Perhitungan Pusat Massa 5](#_Toc171883684)

[2.5 Hasil Menjalankan Menu Perhitungan Kecepatan Pusat Massa 5](#_Toc171883685)

[BAB III – UJI COBA PROGRAM 6](#_Toc171883686)

[3.1 Perhitungan Pusat Massa 6](#_Toc171883687)

[3.1.1 Contoh Soal 6](#_Toc171883688)

[3.1.2 Perhitungan Secara Manual 6](#_Toc171883689)

[3.1.3 Perhitungan Menggunakan Program 6](#_Toc171883690)

[3.2 Perhitungan Kecepatan Pusat Massa 9](#_Toc171883691)

[3.2.1 Contoh Soal 9](#_Toc171883692)

[3.2.2 Perhitungan Secara Manual 9](#_Toc171883693)

[3.2.3 Perhitungan Menggunakan Program 9](#_Toc171883694)

[BAB IV - SOURCE CODE 12](#_Toc171883695)

[BAB V – KESIMPULAN 15](#_Toc171883696)

# BAB I – DASAR TEORI

## Pusat Massa

Pusat massa adalah titik hipotetis dalam sebuah benda atau sistem benda di mana total massa dianggap terkonsentrasi. Dalam fisika, pusat massa sering kali digunakan untuk menyederhanakan analisis gerak, karena gerakan seluruh benda dapat direpresentasikan dengan gerakan titik ini.

Rumus persamaan pusat massa:

A math equation on a black background

Description automatically generated

Keterangan:

A black background with white text

Description automatically generated

## 1.2 Kecepatan Pusat Massa

Kecepatan pusat massa adalah vektor kecepatan yang menggambarkan kecepatan gerakan dari pusat massa suatu sistem benda atau partikel. Ini adalah vektor yang menunjukkan seberapa cepat dan dalam arah mana pusat massa sistem tersebut bergerak.

Rumus persamaan kecepatan pusat massa:

A black background with white text

Description automatically generated

Keterangan:

A black background with white text

Description automatically generated

# BAB II - DEMO APLIKASI

## 2.1 Link Video Demo Aplikasi

<https://youtu.be/vcoB9M6bHXw>

## 2.2 Link Google Colab

<https://colab.research.google.com/drive/1fT-p60rKa2hoE9BJUU1_uBJVyTv30OWV?usp=sharing>

## 2.3 Link GitHub

<https://github.com/weztcy/Program-Python-Center-of-Mass-Velocity>

## 2.4 Code Program Menggunakan Google Collab

A screenshot of a computer program

Description automatically generated

## 2.5 Hasil Menjalankan Menu Perhitungan Pusat Massa

A screenshot of a computer program

Description automatically generated

## 2.5 Hasil Menjalankan Menu Perhitungan Kecepatan Pusat Massa

A screenshot of a computer program

Description automatically generated

# BAB III – UJI COBA PROGRAM

## 3.1 Perhitungan Pusat Massa

### 3.1.1 Contoh Soal

A screen shot of a computer

Description automatically generated

### 3.1.2 Perhitungan Secara Manual

Tabel Data

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Objek | Massa (KG) | Koordinat X (CM) | Koordinat Y (CM) | Koordinat Z (CM) |
| 1 | 1.2 | 0 | 0 | 0 |
| 2 | 2.5 | 140 | 0 | 0 |
| 3 | 3.4 | 70 | 120 | 0 |

Karena ketiga objek tersebut membentuk segitiga sama sisi maka dapat disimpulkan kalau ketiga objek tersebut berada pada dua dimensi saja yaitu X dan Y.

Masukkan ke dalam rumus, dan melakukan perhitungan sebagai berikut:

A math equations on a blue background

Description automatically generated

### 3.1.3 Perhitungan Menggunakan Program

Menjalankan program dan memilih menu nomor 1 yaitu ”Hitung Pusat Massa”.

A screenshot of a computer error

Description automatically generated

Mengisi input sesuai data yang ingin dihitung.

A screenshot of a computer

Description automatically generated

Menjalankan program.

A screenshot of a computer program

Description automatically generated

Hasil perhitungan pusat massa:

* Sumbu X = 82.81690141
* Sumbu Y = 57.46478873
* Sumbu Z = 0

Dapat dilihat pada nilai Sumbu X = 83cm (82.8169014084507 digenapkan), Sumbu Y = 58cm (57.46478873239437 digenapkan), Sumbu Z = 0cm (karena segitiga sama sisi hanya dua dimensi). Maka dari itu disimpulkan hasil pengujian perhitungan pusat massa sudah memiliki akurasi yang tinggi.

Hasil visualisasi sumbu:

* Visualisasi 2D

A graph with colored dots and numbers

Description automatically generated

* Visualisasi 3D

A graph of a graph with numbers and points

Description automatically generated with medium confidence

## 3.2 Perhitungan Kecepatan Pusat Massa

### 3.2.1 Contoh Soal

A black background with white text

Description automatically generated

### 3.2.2 Perhitungan Secara Manual

Tabel Data

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Objek | Massa (KG) | Kecepatan X (m/s) | Kecepatan Y (m/s) | Kecepatan Z (m/s) |
| 1 | 4 | 3 | 0 | 0 |
| 2 | 2 | -1 | 0 | 0 |
| 3 | 3 | 2 | 0 | 0 |

Pada soal, kecepatan objek-objek hanya dinyatakan sepanjang sumbu x, dan tidak ada Sumbu kecepatan yang disebutkan untuk sumbu y atau z. Oleh karena itu, untuk kecepatan y dan z, nilai-nilainya diasumsikan sebagai 0 m/s, mengindikasikan bahwa objek-objek tersebut tidak memiliki gerakan vertikal atau dalam arah kedalaman.

Masukkan ke dalam rumus, dan melakukan perhitungan sebagai berikut:

A black background with white text

Description automatically generated

### 3.2.3 Perhitungan Menggunakan Program

Menjalankan program dan memilih menu nomor 2 yaitu ”Hitung Kecepatan Pusat Massa”.

A screenshot of a computer

Description automatically generated

Mengisi input sesuai data yang ingin dihitung.

A screenshot of a computer

Description automatically generated

Menjalankan program.

A screenshot of a computer

Description automatically generated

Hasil perhitungan kecepatan pusat massa:

* Sumbu X = 1.78
* Sumbu Y = 0
* Sumbu Z = 0

Dapat dilihat pada nilai Sumbu X = 1.78cm (1.77777778 digenapkan), Sumbu Y = 0cm, Sumbu Z = 0cm (karena kecepatan hanya pada satu dimensi). Maka dari itu disimpulkan hasil pengujian perhitungan kecepatan pusat massa sudah memiliki akurasi yang tinggi.

Hasil visualisasi sumbu:

* Visualisasi 2D

A graph with numbers and colored lines

Description automatically generated

* Visualisasi 3D

A graph of a graph with lines and numbers

Description automatically generated

# BAB IV - SOURCE CODE

Center of Mass + Velocity.ipynb

import numpy as np

import matplotlib.pyplot as plt

from mpl\_toolkits.mplot3d import Axes3D

def hitung\_pusat\_massa(posisi, massa):

    total\_massa = np.sum(massa)

    pusat\_massa = np.sum(posisi.T \* massa, axis=1) / total\_massa

    return pusat\_massa

def hitung\_kecepatan\_pusat\_massa(kecepatan, massa):

    total\_massa = np.sum(massa)

    kecepatan\_pusat\_massa = np.sum(kecepatan.T \* massa, axis=1) / total\_massa

    return kecepatan\_pusat\_massa

def visualisasi\_2d(posisi, pusat\_massa, judul):

    plt.figure()

    warna = plt.cm.viridis(np.linspace(0, 1, len(posisi)))

    for i, pos in enumerate(posisi):

        plt.scatter(pos[0], pos[1], color=warna[i], label=f'Objek {i+1}')

    plt.scatter(pusat\_massa[0], pusat\_massa[1], color='red', label='Pusat Massa', edgecolor='black')

    plt.xlabel('X')

    plt.ylabel('Y')

    plt.title(judul)

    plt.legend()

    plt.grid(True)

    plt.show()

def visualisasi\_3d(posisi, pusat\_massa, judul):

    fig = plt.figure()

    ax = fig.add\_subplot(111, projection='3d')

    warna = plt.cm.viridis(np.linspace(0, 1, len(posisi)))

    for i, pos in enumerate(posisi):

        ax.scatter(pos[0], pos[1], pos[2], color=warna[i], label=f'Objek {i+1}')

    ax.scatter(pusat\_massa[0], pusat\_massa[1], pusat\_massa[2], color='red', label='Pusat Massa', edgecolor='black')

    ax.set\_xlabel('X')

    ax.set\_ylabel('Y')

    ax.set\_zlabel('Z')

    ax.set\_title(judul)

    ax.legend()

    plt.show()

def visualisasi\_kecepatan\_2d(kecepatan, kecepatan\_pusat\_massa, judul):

    plt.figure()

    warna = plt.cm.viridis(np.linspace(0, 1, len(kecepatan)))

    for i, kecep in enumerate(kecepatan):

        plt.quiver(0, 0, kecep[0], kecep[1], color=warna[i], scale=1, scale\_units='xy', angles='xy', label=f'Objek {i+1}')

    plt.quiver(0, 0, kecepatan\_pusat\_massa[0], kecepatan\_pusat\_massa[1], color='red', scale=1, scale\_units='xy', angles='xy', label='Kecepatan Pusat Massa')

    plt.xlabel('Vx')

    plt.ylabel('Vy')

    plt.title(judul)

    plt.legend()

    plt.grid(True)

    plt.show()

def visualisasi\_kecepatan\_3d(kecepatan, kecepatan\_pusat\_massa, judul):

    fig = plt.figure()

    ax = fig.add\_subplot(111, projection='3d')

    warna = plt.cm.viridis(np.linspace(0, 1, len(kecepatan)))

    for i, kecep in enumerate(kecepatan):

        ax.quiver(0, 0, 0, kecep[0], kecep[1], kecep[2], color=warna[i], label=f'Objek {i+1}')

    ax.quiver(0, 0, 0, kecepatan\_pusat\_massa[0], kecepatan\_pusat\_massa[1], kecepatan\_pusat\_massa[2], color='red', label='Kecepatan Pusat Massa')

    ax.set\_xlabel('Vx')

    ax.set\_ylabel('Vy')

    ax.set\_zlabel('Vz')

    ax.set\_title(judul)

    ax.legend()

    plt.show()

def main():

    while True:

        print("Menu:")

        print("1. Hitung Pusat Massa")

        print("2. Hitung Kecepatan Pusat Massa")

        print("3. Keluar")

        pilihan = input("Pilih opsi (1/2/3): ")

        if pilihan == '1':

            n = int(input("Masukkan jumlah objek: "))

            massa = []

            posisi = []

            for i in range(n):

                m = float(input(f"Masukkan massa untuk objek {i+1}: "))

                x = float(input(f"Masukkan koordinat x untuk objek {i+1}: "))

                y = float(input(f"Masukkan koordinat y untuk objek {i+1}: "))

                z = float(input(f"Masukkan koordinat z untuk objek {i+1}: "))

                massa.append(m)

                posisi.append([x, y, z])

            posisi = np.array(posisi)

            massa = np.array(massa)

            pusat\_massa = hitung\_pusat\_massa(posisi, massa)

            print(f"Pusat Massa: {pusat\_massa}")

            visualisasi\_2d(posisi, pusat\_massa, "Pusat Massa 2D")

            visualisasi\_3d(posisi, pusat\_massa, "Pusat Massa 3D")

        elif pilihan == '2':

            n = int(input("Masukkan jumlah objek: "))

            massa = []

            kecepatan = []

            for i in range(n):

                m = float(input(f"Masukkan massa untuk objek {i+1}: "))

                vx = float(input(f"Masukkan kecepatan x untuk objek {i+1}: "))

                vy = float(input(f"Masukkan kecepatan y untuk objek {i+1}: "))

                vz = float(input(f"Masukkan kecepatan z untuk objek {i+1}: "))

                massa.append(m)

                kecepatan.append([vx, vy, vz])

            kecepatan = np.array(kecepatan)

            massa = np.array(massa)

            kecepatan\_pusat\_massa = hitung\_kecepatan\_pusat\_massa(kecepatan, massa)

            print(f"Kecepatan Pusat Massa: {kecepatan\_pusat\_massa}")

            visualisasi\_kecepatan\_2d(kecepatan, kecepatan\_pusat\_massa, "Kecepatan Pusat Massa 2D")

            visualisasi\_kecepatan\_3d(kecepatan, kecepatan\_pusat\_massa, "Kecepatan Pusat Massa 3D")

        elif pilihan == '3':

            print("Terima kasih! Program selesai.")

            break

        else:

            print("Opsi tidak valid. Silakan coba lagi.")

if \_\_name\_\_ == "\_\_main\_\_":

    main()

# BAB V – KESIMPULAN

Dokumentasi program "Center of Mass + Velocity" ini menjelaskan tentang sebuah program Python yang dirancang untuk menghitung pusat massa dan kecepatan pusat massa dari sejumlah objek, serta memvisualisasikan hasil perhitungannya. Program ini memanfaatkan pustaka NumPy untuk perhitungan numerik dan Matplotlib untuk visualisasi. Tujuan utama dari program ini adalah untuk menyederhanakan analisis gerak sistem benda dengan memusatkan massa dan kecepatan ke dalam satu titik hipotetis yang disebut pusat massa. Kecepatan pusat massa adalah vektor yang menunjukkan seberapa cepat dan dalam arah mana pusat massa sistem tersebut bergerak.

Program ini menawarkan dua fungsi utama, yaitu menghitung pusat massa dan menghitung kecepatan pusat massa. Pengguna diminta untuk memasukkan data massa serta posisi atau kecepatan objek-objek, kemudian program melakukan perhitungan dan menampilkan hasilnya dalam bentuk angka dan grafik 2D serta 3D. Dokumentasi ini juga mencakup dasar teori tentang pusat massa dan kecepatan pusat massa, yang membantu pengguna memahami konsep di balik perhitungan yang dilakukan.

Uji coba program dilakukan dengan beberapa contoh soal yang mencakup perhitungan manual dan perhitungan menggunakan program. Hasil dari uji coba ini menunjukkan bahwa program memiliki akurasi tinggi dalam menghitung pusat massa dan kecepatan pusat massa. Hasil perhitungan yang divisualisasikan dalam grafik 2D dan 3D membantu pengguna memahami distribusi massa dan gerakan dari sistem benda yang dianalisis. Dokumentasi ini secara keseluruhan menunjukkan bahwa program ini merupakan alat yang bermanfaat untuk menganalisis pusat massa dan kecepatan pusat massa dalam sistem multi-objek, serta memastikan bahwa hasil yang diberikan akurat dan dapat diandalkan.