

1. Menu 1 : Pilihan mengenai apa yang ingin dihitung.
2. Input : 1

Menghitung Massa

```
dz — Python D121211002_Tugas2_MKN.py — 65x13
isriany-MacBook-Air:dz isriany$ python3 D121211002_Tugas2_MKN.py

Parachutist Case

Please choose what you want to calculate:
1. Mass (m)
2. Drag Coefficient (c)
3. Velocity (vt)
4. Time (t)
5. Exit

Your choice:
1
```

3. Memasukkan nilai- nilai yang diminta.

Input :

- Velocity : 45
- Time : 15
- Drag Coefficient : 25
- Approximate Error: 0.1
- Lower bound (x_l) : 80
- Upper bound (x_u) : 190

```
dz — Python D121211002_Tugas2_MKN.py — 65x25

Parachutist Case

Please choose what you want to calculate:
1. Mass (m)
2. Drag Coefficient (c)
3. Velocity (vt)
4. Time (t)
5. Exit

Your choice:
1

Please input the velocity (vt): 45

Please input the time (t): 15

Please input the drag coefficient (c): 25

Please input the approximate error (es): 0.1

Please input the lower bound (xl): 80

Please input the upper bound (xu): 190
```

4. Menu 2 :Pilihan mengenai metode yang akan digunakan.

Input: 3

```
dz — Python D121211002_Tugas2_MK...

Please choose what method you want to use:
1. Bisection Method
2. False-Position Method
3. Bisection and False-Position Method

Your choice:
3
```

5. Hasil Perhitungan

```
dz — Python D121211002_Tugas2_MKN.py — 94x28

Compute Mass with Bisection Method
=====
| Iteration |   xl   |   fxl   |   xu   |   fxu   |   xr   |   fxr   |   ea   |
=====
| 1         | 80.0000 | -13.9288 | 190.0000 | 19.13145 | 135.00000 | 4.62962 | 25.581 |
| 2         | 80.0000 | -13.9288 | 135.0000 | 4.62962 | 107.50000 | -4.14740 | 11.34  |
| 3         | 107.5000 | -4.14740 | 135.0000 | 4.62962 | 121.25000 | 0.37330 | 6.011  |
| 4         | 107.5000 | -4.14740 | 121.2500 | 0.37330 | 114.37500 | -1.85428 | 2.918  |
| 5         | 114.3750 | -1.85428 | 121.2500 | 0.37330 | 117.81250 | -0.73223 | 1.438  |
| 6         | 117.8125 | -0.73223 | 121.2500 | 0.37330 | 119.53125 | -0.17739 | 0.714  |
| 7         | 119.53125 | -0.17739 | 121.2500 | 0.37330 | 120.39062 | 0.09847 | 0.358  |
| 8         | 119.53125 | -0.17739 | 120.39062 | 0.09847 | 119.96094 | -0.03933 | 0.179  |
| 9         | 119.96094 | -0.03933 | 120.39062 | 0.09847 | 120.17578 | 0.02960 | 0.089  |
| 10        | 119.96094 | -0.03933 | 120.17578 | 0.02960 | 120.06836 | -0.00486 | 0.089  |
=====
So, mass of parachutist is 120.068

Compute Mass with False-Position Method
=====
| Iteration |   xl   |   fxl   |   xu   |   fxu   |   xr   |   fxr   |   ea   |
=====
| 1         | 80.000  | -13.929 | 190.000 | 19.131  | 126.345 | 1.981   | 4.787  |
| 2         | 80.000  | -13.929 | 126.345 | 1.981   | 120.573 | 1.981   | 0.376  |
| 3         | 80.000  | -13.929 | 120.573 | 0.157   | 120.121 | 0.157   | 0.029  |
| 4         | 80.000  | -13.929 | 120.121 | 0.012   | 120.086 | 0.012   | 0.029  |
=====
So, mass of parachutist is 120.086
```

6. Pilihan untuk menghitung kembali

Input : n

```
dz — -bash — 37x5

Want to calculate again? (y/n): n
End of Program

isrianys-MacBook-Air:dz isrianys$
```

7. Tampilan keseluruhan

```
dz — bash — 116x64

Parachutist Case

Please choose what you want to calculate:
1. Mass (m)
2. Drag Coefficient (c)
3. Velocity (vt)
4. Time (t)
5. Exit

Your choice:
1

Please input the velocity (vt): 45

Please input the time (t): 15

Please input the drag coefficient (c): 25

Please input the approximate error (es): 0.1

Please input the lower bound (xl): 80

Please input the upper bound (xu): 190

Please choose what method you want to use:
1. Bisection Method
2. False-Position Method
3. Bisection and False-Position Method

Your choice:
3

Compute Mass with Bisection Method
=====
| Iteration | xl      | fxl     | xu      | fxu     | xr      | fxr     | ea      |
|=====|=====|=====|=====|=====|=====|=====|=====|
| 1         | 80.00000 | -13.92882 | 190.00000 | 19.13145 | 135.00000 | 4.62962 | 25.581  |
| 2         | 80.00000 | -13.92882 | 135.00000 | 4.62962 | 107.50000 | -4.14740 | 11.34   |
| 3         | 107.50000 | -4.14740 | 135.00000 | 4.62962 | 121.25000 | 0.37330 | 6.011   |
| 4         | 107.50000 | -4.14740 | 121.25000 | 0.37330 | 114.37500 | -1.85428 | 2.918   |
| 5         | 114.37500 | -1.85428 | 121.25000 | 0.37330 | 117.81250 | -0.73223 | 1.438   |
| 6         | 117.81250 | -0.73223 | 121.25000 | 0.37330 | 119.53125 | -0.17739 | 0.714   |
| 7         | 119.53125 | -0.17739 | 121.25000 | 0.37330 | 120.39062 | 0.09847 | 0.358   |
| 8         | 119.53125 | -0.17739 | 120.39062 | 0.09847 | 119.96094 | -0.03933 | 0.179   |
| 9         | 119.96094 | -0.03933 | 120.39062 | 0.09847 | 120.17578 | 0.02960 | 0.089   |
| 10        | 119.96094 | -0.03933 | 120.17578 | 0.02960 | 120.06836 | -0.00486 | 0.009   |
|=====|=====|=====|=====|=====|=====|=====|=====|

So, mass of parachutist is 120.068

Compute Mass with False-Position Method
=====
| Iteration | xl      | fxl     | xu      | fxu     | xr      | fxr     | ea      |
|=====|=====|=====|=====|=====|=====|=====|=====|
| 1         | 80.000  | -13.929 | 190.000 | 19.131  | 126.345 | 1.981   | 4.787   |
| 2         | 80.000  | -13.929 | 126.345 | 1.981   | 120.573 | 0.157   | 0.376   |
| 3         | 80.000  | -13.929 | 120.573 | 0.157   | 120.121 | 0.012   | 0.029   |
| 4         | 80.000  | -13.929 | 120.121 | 0.012   | 120.086 | 0.012   | 0.029   |
|=====|=====|=====|=====|=====|=====|=====|=====|

So, mass of parachutist is 120.086

Want to calculate again? (y/n): n
End of Program
```