

LAPORAN PRAKTIKUM
ALGORITMA DAN STRUKTUR DATA
MODUL 10
ANALISIS ALGORITMA



Disusun oleh:
Bima Triadmaja
L200210137
E

TEKNIK INFORMATIKA
FAKULTAS KOMUNIKASI DAN INFORMATIKA
UNIVERSITAS MUHAMMADIYAH SURAKARTA
2022/2023

Soal-soal untuk mahasiswa

1. Kerjakan ulang contoh dan latihan di modul ini menggunakan modul timeit.

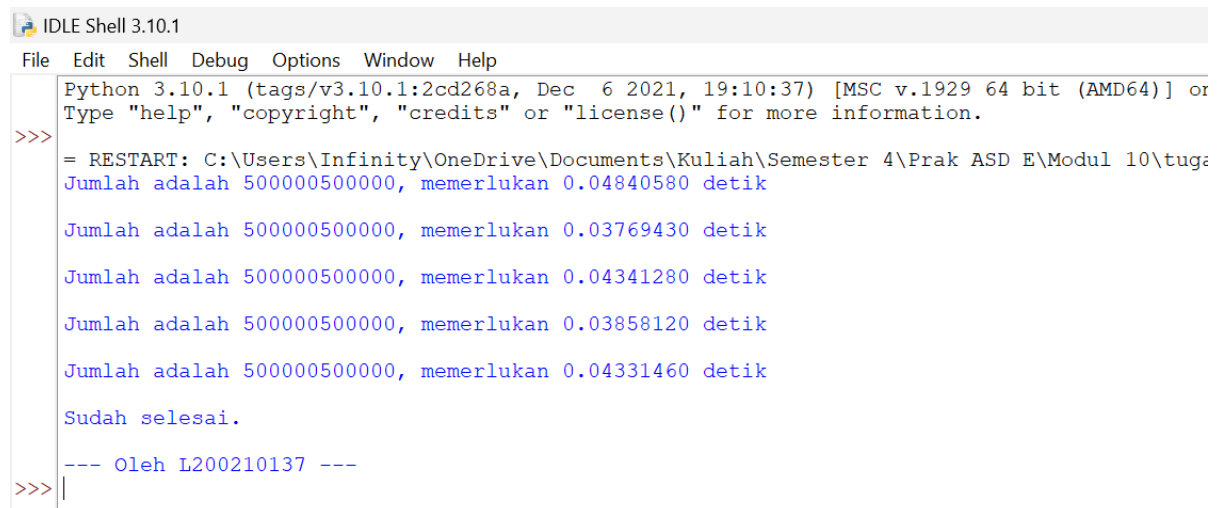
a. jumlahkan_cara_1

```
from timeit import timeit
import random

def jumlahkan_cara_1(n):
    hasilnya = 0
    for i in range(1,n+1):
        hasilnya = hasilnya + i
    return hasilnya

for i in range(5):
    a = jumlahkan_cara_1(1000000)
    b = timeit("jumlahkan_cara_1(1000000)", "from __main__
import jumlahkan_cara_1", number = 1)
    print("Jumlah adalah %d, memerlukan %9.8f detik" %(a,
b))
    print('')
print('Sudah selesai.')

print('\n--- Oleh L200210137 ---')
```



```
IDLE Shell 3.10.1
File Edit Shell Debug Options Window Help
Python 3.10.1 (tags/v3.10.1:2cd268a, Dec 6 2021, 19:10:37) [MSC v.1929 64 bit (AMD64)] or
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:\Users\Infinity\OneDrive\Documents\Kuliah\Semester 4\Prak ASD E\Modul 10\tuga
Jumlah adalah 500000500000, memerlukan 0.04840580 detik

Jumlah adalah 500000500000, memerlukan 0.03769430 detik

Jumlah adalah 500000500000, memerlukan 0.04341280 detik

Jumlah adalah 500000500000, memerlukan 0.03858120 detik

Jumlah adalah 500000500000, memerlukan 0.04331460 detik

Sudah selesai.

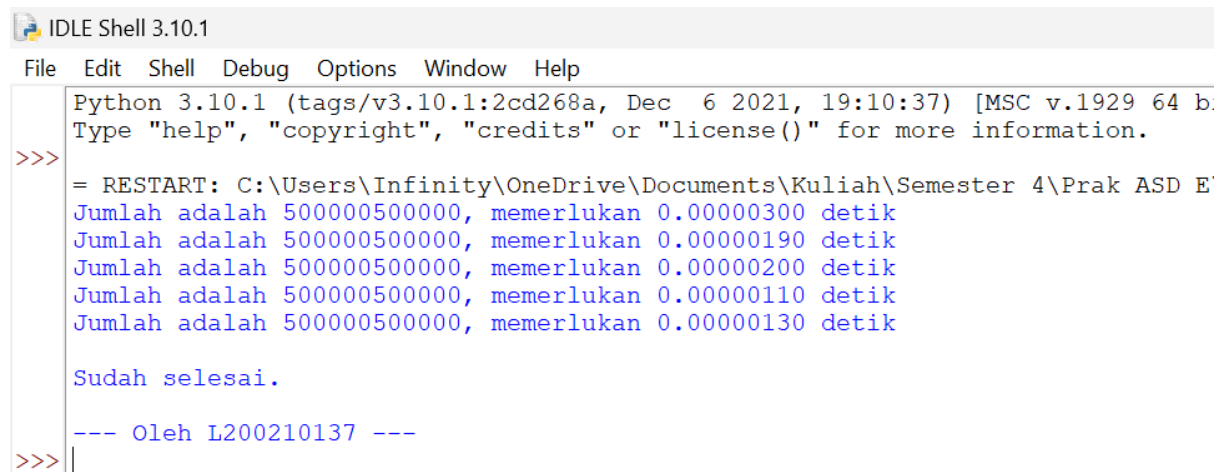
--- Oleh L200210137 ---
>>> |
```

b. jumlahkan_cara_2

```
from timeit import timeit
import random

def jumlahkan_cara_2(n):
    return(n * (n+1))/2

for i in range(5):
    siap = "from __main__ import jumlahkan_cara_2"
    h = jumlahkan_cara_2(1000000)
    t = timeit("jumlahkan_cara_2(1000000)", setup=siap,
number = 1)
    print("Jumlah adalah %d, memerlukan %9.8f detik" %(h,
t))
print('')
print('Sudah selesai.')
print('\n--- Oleh L200210137 ---')
```



```
IDLE Shell 3.10.1
File Edit Shell Debug Options Window Help
Python 3.10.1 (tags/v3.10.1:2cd268a, Dec 6 2021, 19:10:37) [MSC v.1929 64 b.
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:\Users\Infinity\OneDrive\Documents\Kuliah\Semester 4\Prak ASD E'
Jumlah adalah 500000500000, memerlukan 0.00000300 detik
Jumlah adalah 500000500000, memerlukan 0.00000190 detik
Jumlah adalah 500000500000, memerlukan 0.00000200 detik
Jumlah adalah 500000500000, memerlukan 0.00000110 detik
Jumlah adalah 500000500000, memerlukan 0.00000130 detik

Sudah selesai.

--- Oleh L200210137 ---
>>>|
```

c. insertionSort

```
from timeit import timeit
import random

def insertionSort(A):
    n = len(A)
    for i in range(1,n):
        nilai = A[i]
        pos = i
        while pos > 0 and nilai < A[pos - 1]:
            A[pos] = A[pos - 1]
            pos = pos - 1
        A[pos] = nilai
```

```

for i in range(5):
    angka = list (range(3000))
    random.shuffle(angka)
    t = timeit("insertionSort(angka)", "from __main__
import insertionSort, angka", number = 1)
    print("Menggunakan %d bilangan, memerlukan %8.7f
detik" %(len(angka), t))
print('')
print('Sudah selesai.')

print('\n--- Oleh L200210137 ---')

>>> |===== RESTART: C:\Users\Infinity\OneDrive\l
Menggunakan 3000 bilangan, memerlukan 0.0035785 detik
Menggunakan 3000 bilangan, memerlukan 0.0031460 detik
Menggunakan 3000 bilangan, memerlukan 0.0027381 detik
Menggunakan 3000 bilangan, memerlukan 0.0029663 detik
Menggunakan 3000 bilangan, memerlukan 0.0034390 detik

Sudah selesai.

--- Oleh L200210137 ---
>>> |

```

2. Menentukan best case dan average case bagi sorted() serta mengkonfirmasi bahwa data input urutan terbalik bukan kasus terburuk bagi sorted()

```

from timeit import timeit
import random

print("\nHasil sorted average case")

for i in range(5):
    g = list(range(3000))
    random.shuffle(g)
    t = timeit("sorted(g)", setup = "from __main__ import g",
number=1)
    print("Menggunakan %d bilangan, memerlukan %8.7f detik"
%(len(g), t))

print("\nHasil sorted best case")

for i in range(5):
    g = list(range(3000))
    t = timeit("sorted(g)", setup = "from __main__ import g",
number=1)
    print("Menggunakan %d bilangan, memerlukan %8.7f detik"
%(len(g), t))

```

```

print("\nHasil sorted worst case")

for i in range(5):
    g = list(range(3000))
    g = g[::-1]
    t = timeit("sorted(g)", setup = "from __main__ import g",
number=1)
    print("Menggunakan %d bilangan, memerlukan %8.7f detik"
%(len(g), t))
print('')
print('Sudah selesai.')

print('\n--- Oleh L200210137 ---')

```

```

>>> |===== RESTART: C:\Users\Infinity\OneDrive\Documents\Kuliah\
Hasil sorted average case
Menggunakan 3000 bilangan, memerlukan 0.0003378 detik
Menggunakan 3000 bilangan, memerlukan 0.0003270 detik
Menggunakan 3000 bilangan, memerlukan 0.0003196 detik
Menggunakan 3000 bilangan, memerlukan 0.0005035 detik
Menggunakan 3000 bilangan, memerlukan 0.0003738 detik

Hasil sorted best case
Menggunakan 3000 bilangan, memerlukan 0.0000422 detik
Menggunakan 3000 bilangan, memerlukan 0.0000224 detik
Menggunakan 3000 bilangan, memerlukan 0.0000214 detik
Menggunakan 3000 bilangan, memerlukan 0.0000230 detik
Menggunakan 3000 bilangan, memerlukan 0.0000206 detik

Hasil sorted worst case
Menggunakan 3000 bilangan, memerlukan 0.0000288 detik
Menggunakan 3000 bilangan, memerlukan 0.0000270 detik
Menggunakan 3000 bilangan, memerlukan 0.0000277 detik
Menggunakan 3000 bilangan, memerlukan 0.0000248 detik
Menggunakan 3000 bilangan, memerlukan 0.0000253 detik

Sudah selesai.

--- Oleh L200210137 ---
>>> |

```

3. Menentukan running time

- a. Loop di dalam loop, keduanya sebanyak n :

```

import time
import random
import timeit
import matplotlib.pyplot as plt

```

```

def a(n):
    test = 0
    for i in range(n):
        for j in range(n):
            test = test + i * j

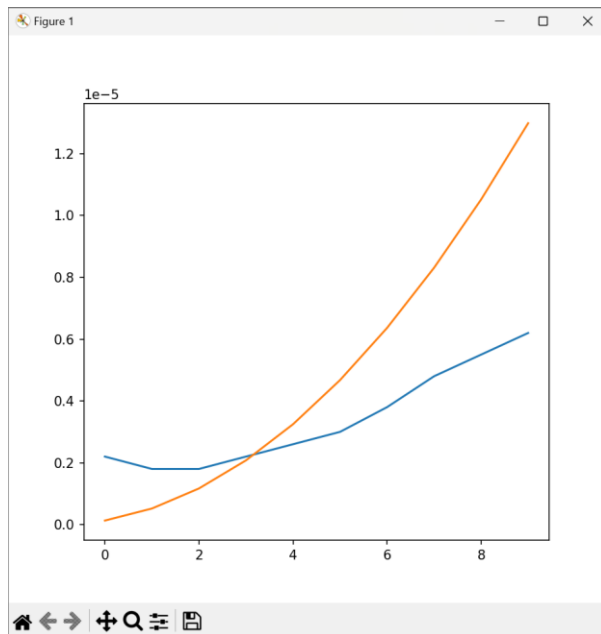
def ujia(n):
    ls=[]
    jangkauan = range(1,n+1)
    siap = "from __main__ import a"
    for i in jangkauan:
        t = timeit.timeit("a(" + str(i) + ")",setup=siap,
number=1)
        ls.append(t)
    return ls

n = 10
LS = ujia(n)

plt.plot(LS)
skala = 7700000
plt.plot([x*x/skala for x in range(1,n+1)])
plt.show()
print('')
print('Sudah selesai.')

print('\n--- Oleh L200210137 ---')

```



b. Loop di dalam loop, yang dalam bergantung nilai i loop luar :

```
import time
import random
import timeit
import matplotlib.pyplot as plt
```

```
def a(n):
    test = 0
    for i in range(n):
        for j in range(n):
            test = test + i * j

def ujia(n):
    ls=[]
    jangkauan = range(1,n+1)
    siap = "from __main__ import a"
    for i in jangkauan:
        t = timeit.timeit("a(" + str(i) + ")",setup=siap,
number=1)
        ls.append(t)
    return ls
```

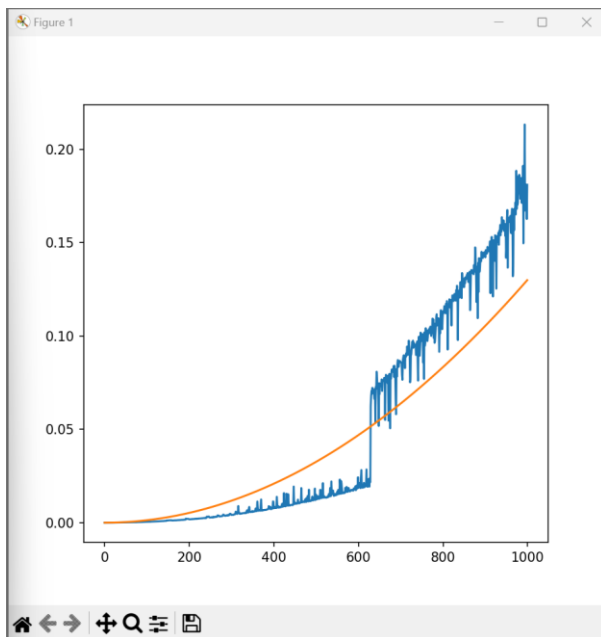
```

n = 1000
LS = ujia(n)

plt.plot(LS)
skala = 7700000
plt.plot([x*x/skala for x in range(1,n+1)])
plt.show()
print('')
print('Sudah selesai.')

print('\n--- Oleh L200210137 ---')

```



c. Dua loop terpisah :

```

import time
import random
import timeit
import matplotlib.pyplot as plt

def a(n):
    test = 0
    for i in range(n):
        for j in range(n):
            test = test + i * j

```



```

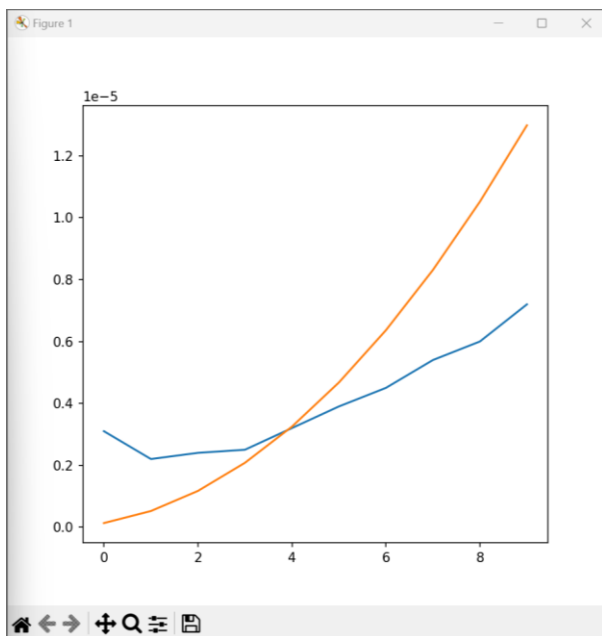
def ujia(n):
    ls=[]
    jangkauan = range(1,n+1)
    siap = "from __main__ import a"
    for i in jangkauan:
        t = timeit.timeit("a(" + str(i) + ")",setup=siap,
number=1)
        ls.append(t)
    return ls

n = 10
LS = ujia(n)

plt.plot(LS)
skala = 7700000
plt.plot([x*x/skala for x in range(1,n+1)])
plt.show()
print('')
print('Sudah selesai.')

print('\n--- Oleh L200210137 ---')

```



d. While loop yang dipangkas separuh tiap putaran :

```

import time
import random
import timeit

```

```

import matplotlib.pyplot as plt

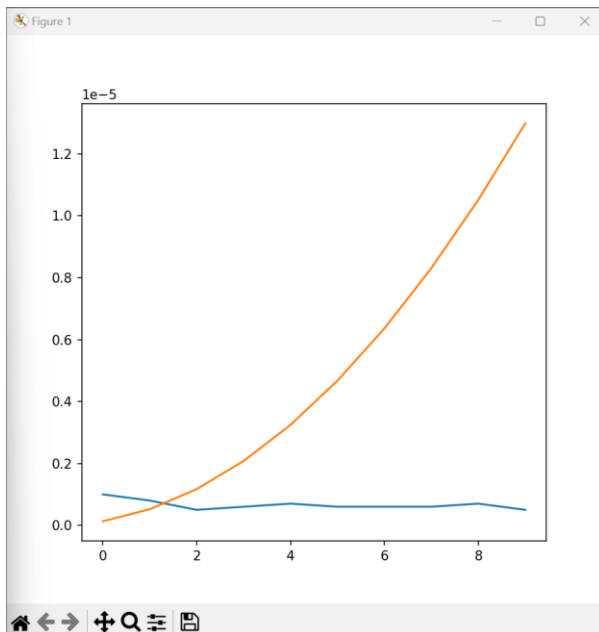
def a(n):
    i == n
    while i > 0:
        k = 2 + 2
        i = i//2

def ujia(n):
    ls=[]
    jangkauan = range(1,n+1)
    siap = "from __main__ import a"
    for i in jangkauan:
        t = timeit.timeit("a(" + str(i) + ")",setup=siap,
number=1)
        ls.append(t)
    return ls

n = 10
LS = ujia(n)

plt.plot(LS)
skala = 7700000
plt.plot([x*x/skala for x in range(1,n+1)])
plt.show()
print('')
print('Sudah selesai.')

print('\n--- Oleh L200210137 ---')
```



e. Loop in a loop, ketiganya sebanyak n :

```
import time
import random
import timeit
import matplotlib.pyplot as plt
```

```
def a(n):
    for i in range(n):
        for j in range(n):
            for k in range(n):
                m = i + j + k + 2019
```

```
def ujia(n):
    ls=[]
    jangkauan = range(1,n+1)
    siap = "from __main__ import a"
    for i in jangkauan:
        t = timeit.timeit("a(" + str(i) + ")",setup=siap,
number=1)
        ls.append(t)
    return ls
```

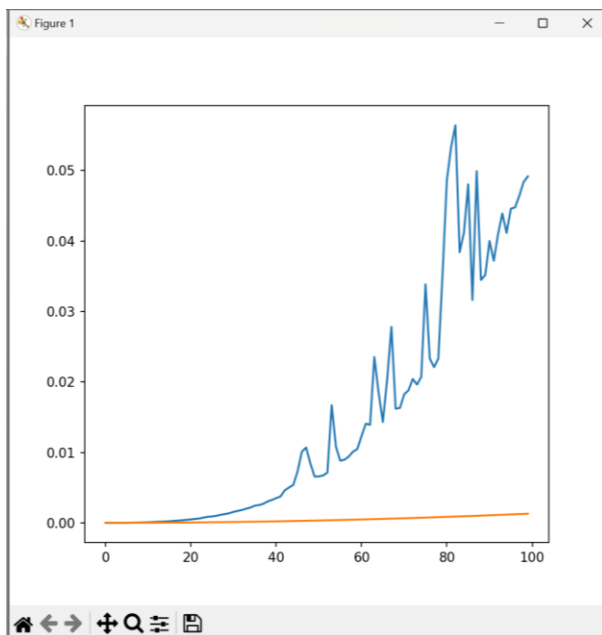
```

n = 100
LS = ujia(n)

plt.plot(LS)
skala = 7700000
plt.plot([x*x/skala for x in range(1,n+1)])
plt.show()
print('')
print('Sudah selesai.')

print('\n--- Oleh L200210137 ---')

```



- f. Loop in a loop in a loop, dengan loop dalam sebanyak nilai loop luar terdekat :

```

import time
import random
import timeit
import matplotlib.pyplot as plt

def a(n):
    for i in range(n):
        if i % 3 == 0:
            for j in range(n//2):

```

```

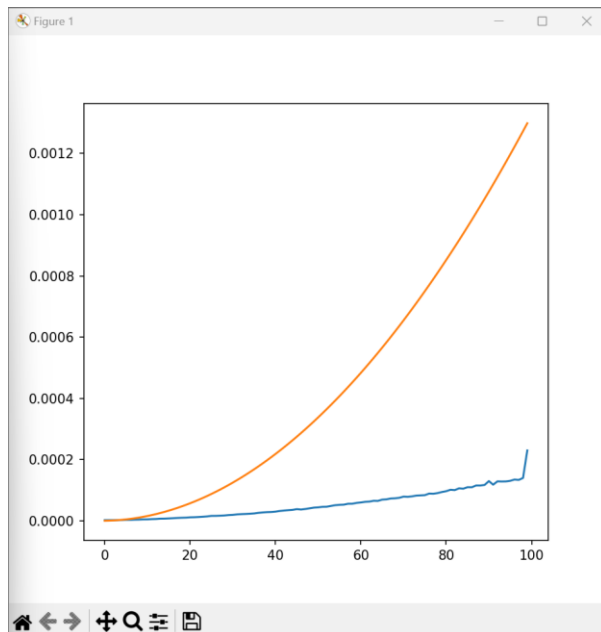
        j += j
    elif i % 2 == 0:
        for j in range(5):
            j += j
    else:
        for j in range(n):
            j += j

def ujia(n):
    ls=[]
    jangkauan = range(1,n+1)
    siap = "from __main__ import a"
    for i in jangkauan:
        t = timeit.timeit("a(" + str(i) + ")",setup=siap,
number=1)
        ls.append(t)
    return ls

n = 100
LS = ujia(n)

plt.plot(LS)
skala = 7700000
plt.plot([x*x/skala for x in range(1,n+1)])
plt.show()
print('')
print('Sudah selesai.')

print('\n--- Oleh L200210137 ---')
```



g. Fungsi :

```
import time
import random
import timeit
import matplotlib.pyplot as plt
```

```
def a(n):
    sum = 0
    for i in range(n):
        if i % 3 == 0:
            for j in range(n//2):
                sum += j
        elif i % 2 == 0:
            for j in range(5):
                sum += j
        else:
            for j in range(n):
                sum += j
```

```
def ujia(n):
    ls=[]
    jangkauan = range(1,n+1)
```

```

    siap = "from __main__ import a"
    for i in jangkauan:
        t = timeit.timeit("a(" + str(i) + ")", setup=siap,
number=1)
        ls.append(t)
    return ls

```

```

n = 100
LS = ujia(n)

```

```

plt.plot(LS)
skala = 7700000
plt.plot([x*x/skala for x in range(1,n+1)])
plt.show()
print('')
print('Sudah selesai.')

```

```

print('\n--- Oleh L200210137 ---')

```

